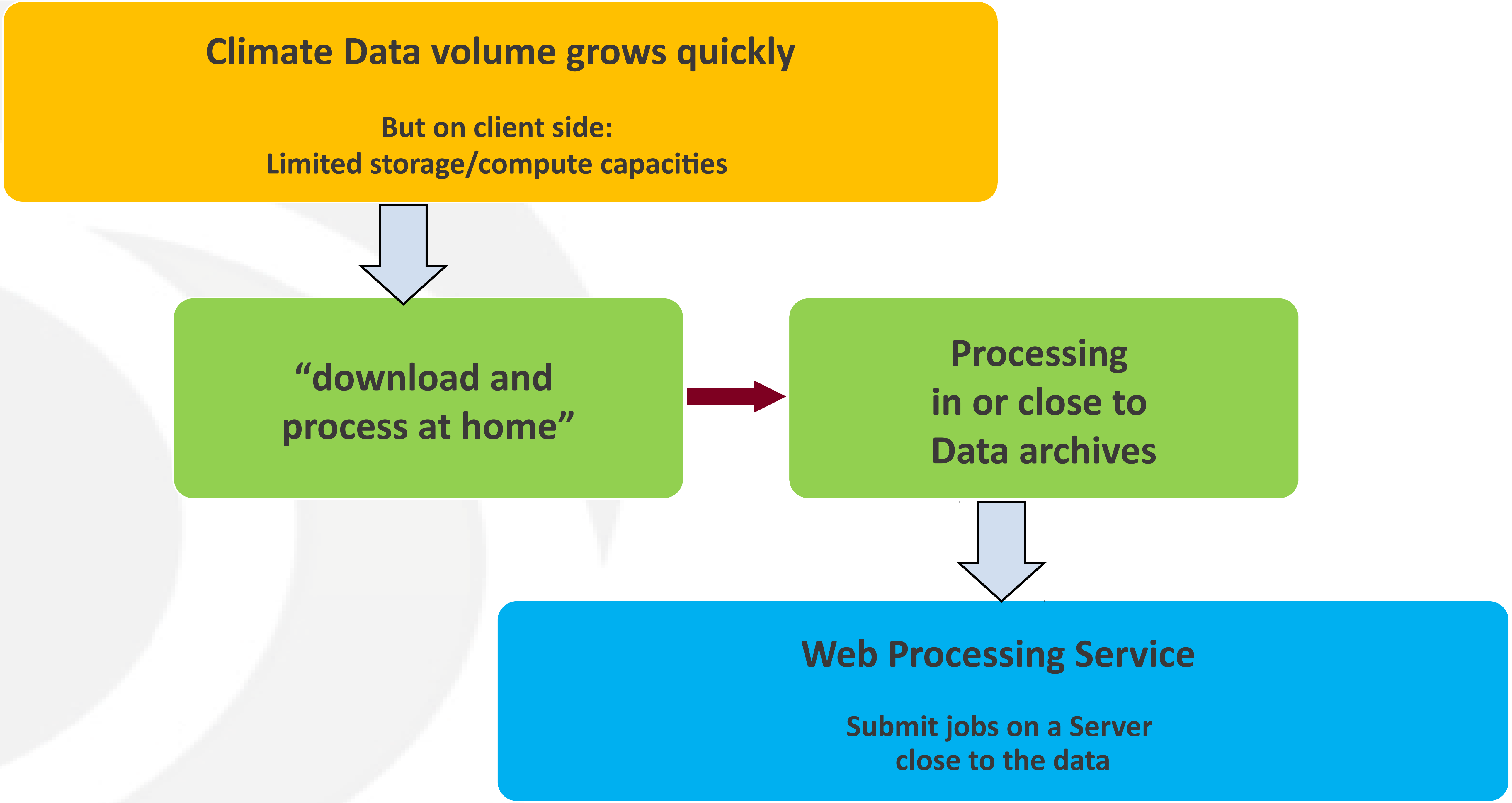
birdhouse:

a collection of web processing services for climate data

Carsten Ehbrecht[1], Nils Hempelmann[2] et. al.

1. German Climate Computing Center, Germany
2. Le Laboratoire des Sciences du Climat et de l'Environnement, France

DKRZ cnrs



Climate Data volume grows quickly

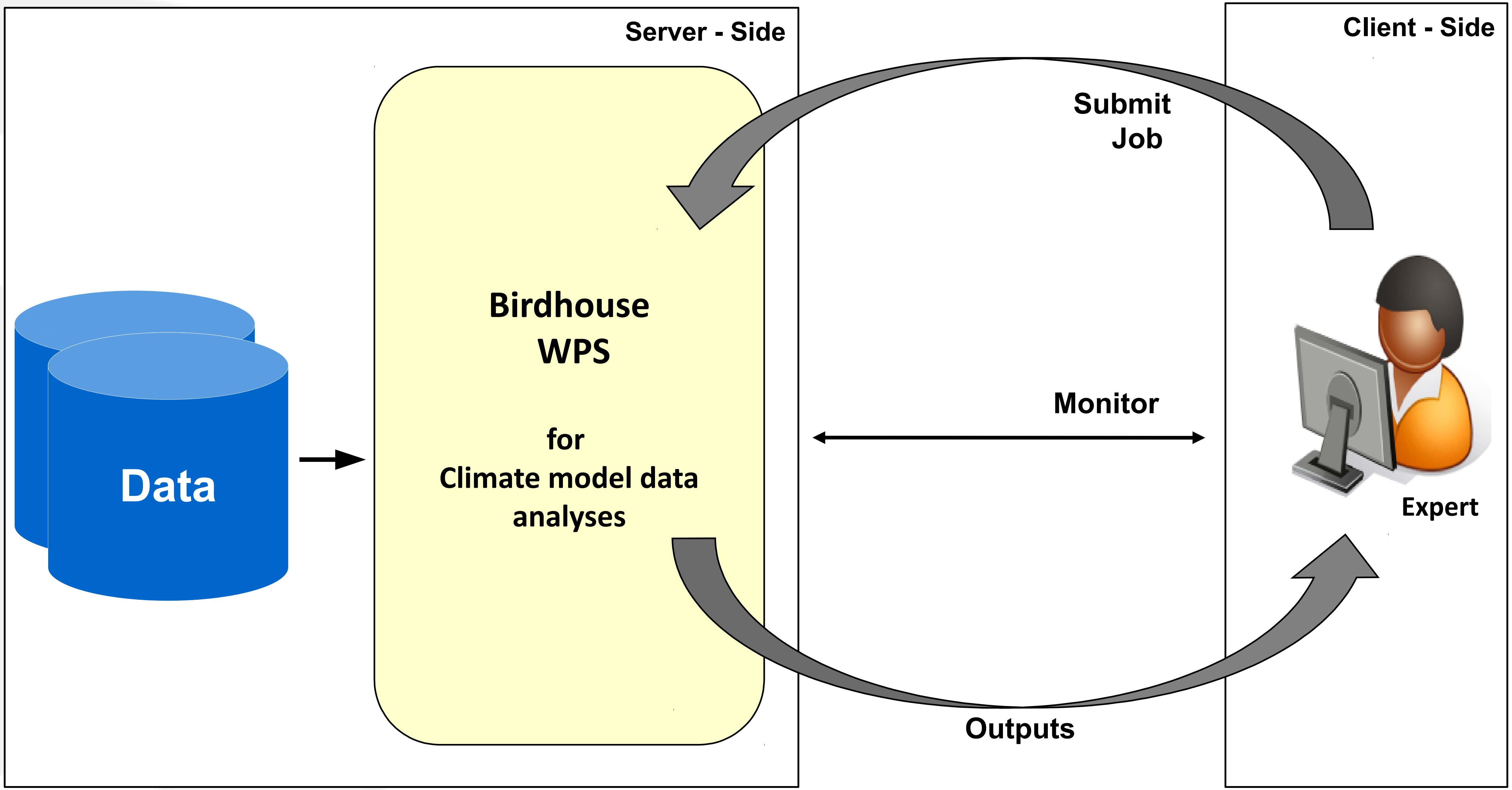
But on client side:
Limited storage/compute capacities

"download and process at home"

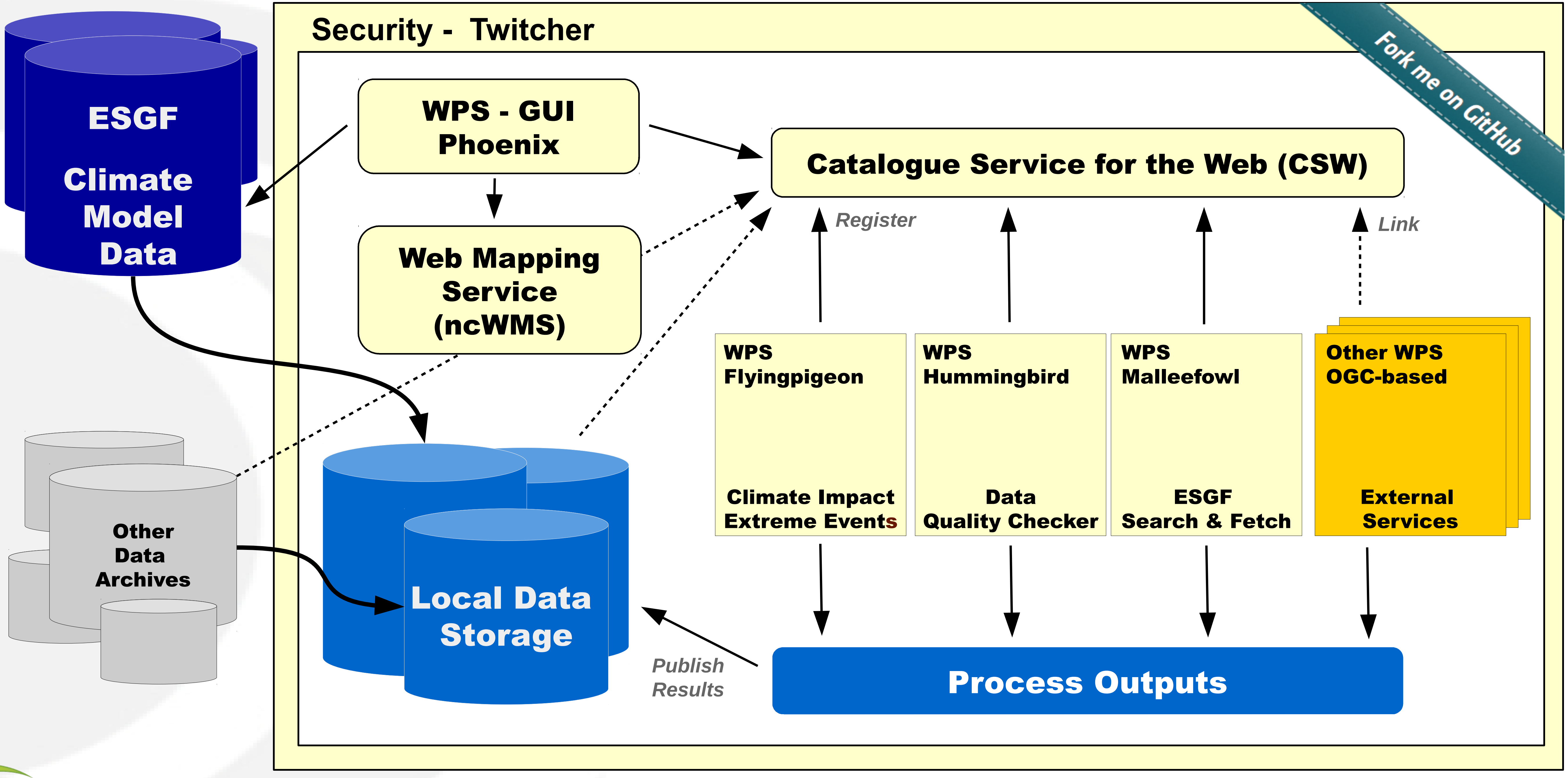
Processing in or close to Data archives

Web Processing Service

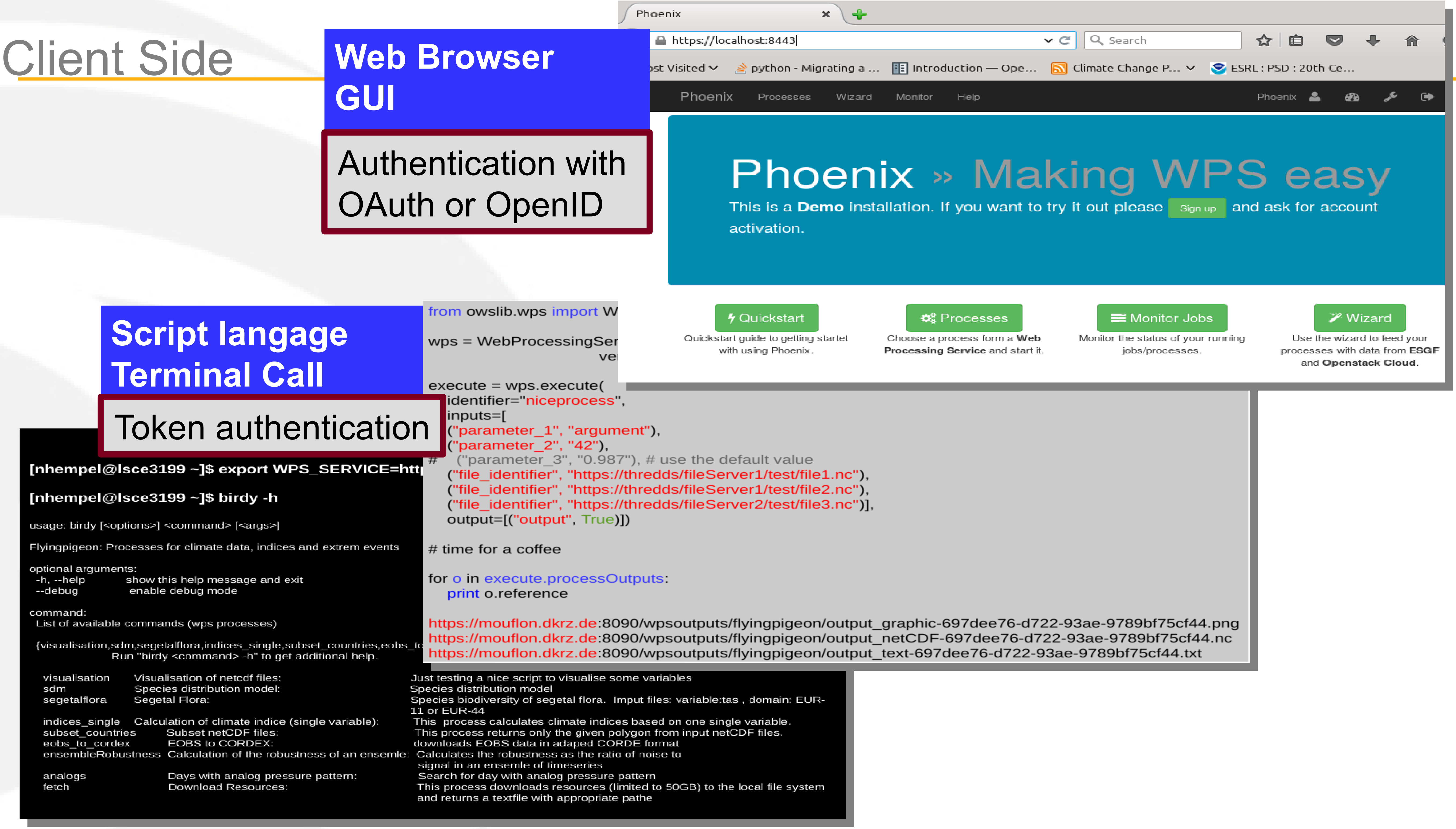
Submit jobs on a Server
close to the data


Server - Side
Client - Side
Submit Job
Birdhouse WPS
for Climate model data analyses
Data
Monitor
Expert
Outputs

Birdhouse - Ecosystem



Client Side



Web Browser GUI

Authentication with OAuth or OpenID

Phoenix » Making WPS easy

This is a **Demo** installation. If you want to try it out please Sign up and ask for account activation.

Quickstart guide to getting startet with using Phoenix.

Choose a process form a **Web Processing Service** and start it.

Monitor the status of your running jobs/processes.

Use the wizard to feed your processes with data from **ESGF** and **Openstack Cloud**.

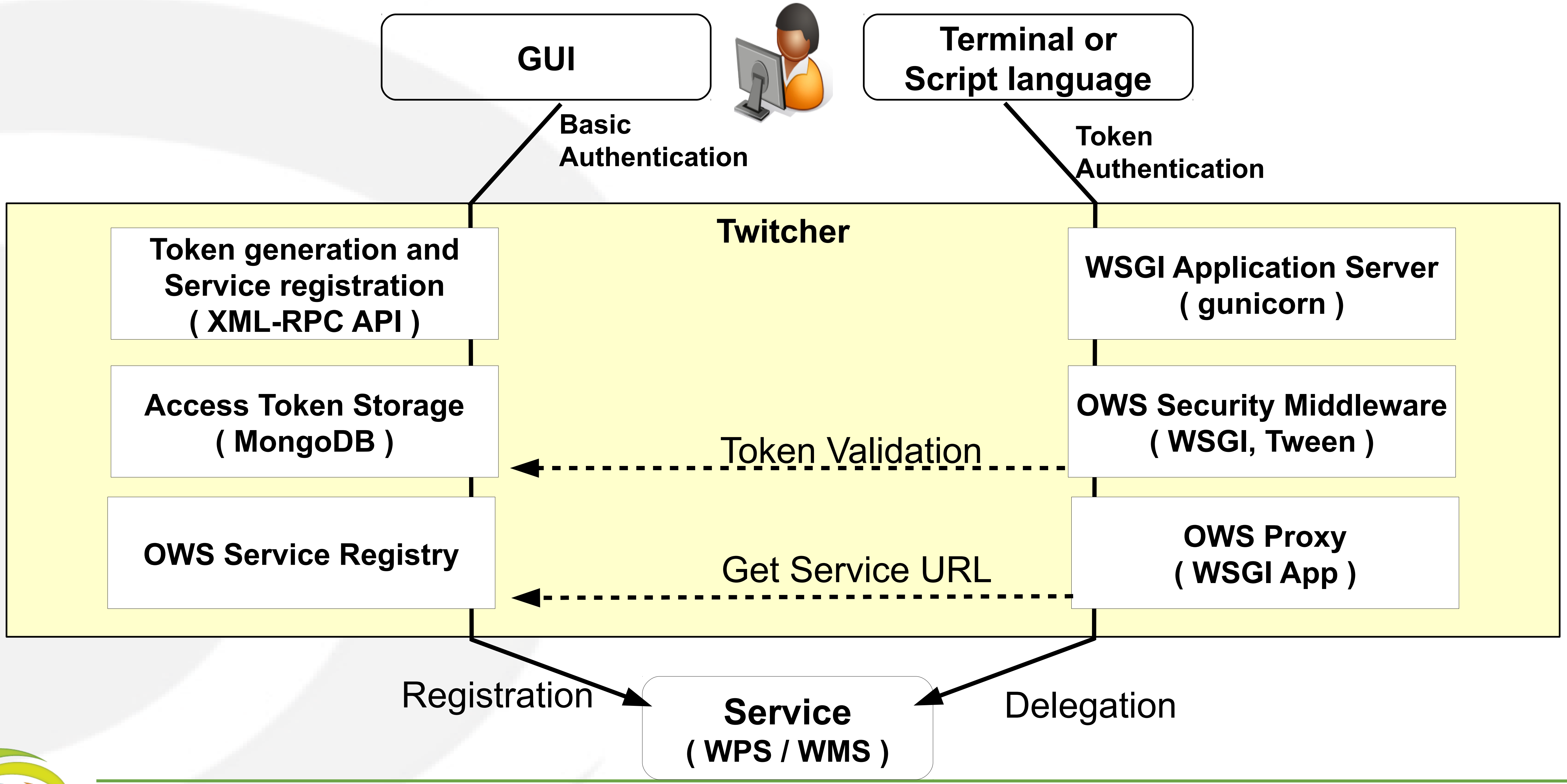
Script langage Terminal Call

Token authentication

```
from owslib.wps import W
wps = WebProcessingSer
                     ve

execute = wps.execute(
    identifier="niceprocess",
    inputs=[
    ("parameter_1", "argument"),
    ("parameter_2", "42"),
#     ("parameter_3", "0.987"), # use the default value
    ("file_identifier", "https://thredds/fileServer1/test/file1.nc"),
    ("file_identifier", "https://thredds/fileServer1/test/file2.nc"),
    ("file_identifier", "https://thredds/fileServer2/test/file3.nc")],
    output=[("output", True)])

# time for a coffee

for o in execute.processOutputs:
    print o.reference

https://mouflon.dkrz.de:8090/wpsoutputs/flyingpigeon/output_graphic-697dee76-d722-93ae-9789bf75cf44.png
https://mouflon.dkrz.de:8090/wpsoutputs/flyingpigeon/output_netCDF-697dee76-d722-93ae-9789bf75cf44.nc
https://mouflon.dkrz.de:8090/wpsoutputs/flyingpigeon/output_text-697dee76-d722-93ae-9789bf75cf44.txt
```

```
[nhempel@lsce3199 ~]$ export WPS_SERVICE=htt

[nhempel@lsce3199 ~]$ birdy -h

usage: birdy [<options>] <command> [<args>]

Flyingpigeon: Processes for climate data, indices and extrem events

optional arguments:
  -h, --help            show this help message and exit
  --debug               enable debug mode

command:
  List of available commands (wps processes)

  {visualisation,sdm,segetalflora,indices_single,subset_countries,eobs_to
                     Run "birdy <command> -h" to get additional help.

    visualisation      Visualisation of netcdf files:                      Just testing a nice script to visualise some variables
    sdm                Species distribution model:                         Species distribution model
    segetalflora       Segetal Flora:                                      Species biodiversity of segetal flora.  Imput files: variable:tas , domain: EUR-
                                                                           11 or EUR-44
    indices_single     Calculation of climate indice (single variable):    This  process calculates climate indices based on one single variable.
    subset_countries         Subset netCDF files:                          This process returns only the given polygon from input netCDF files.
    eobs_to_cordex           EOBS to CORDEX:                               downloads EOBS data in adaped CORDE format
    ensembleRobustness  Calculation of the robustness of an ensemle:       Calculates the robustness as the ratio of noise to
                                                                            signal in an ensemle of timeseries
    analogs                  Days with analog pressure pattern:             Search for day with analog pressure pattern
    fetch                    Download Resources:                            This process downloads resources (limited to 50GB) to the local file system
                                                                            and returns a textfile with appropriate pathe
```

Security



GUI

Terminal or Script language

Basic Authentication

Token Authentication

Twitcher

Token generation and Service registration (XML-RPC API)

Access Token Storage (MongoDB)

OWS Service Registry

WSGI Application Server (gunicorn)

OWS Security Middleware (WSGI, Tween)

OWS Proxy (WSGI App)

Token Validation

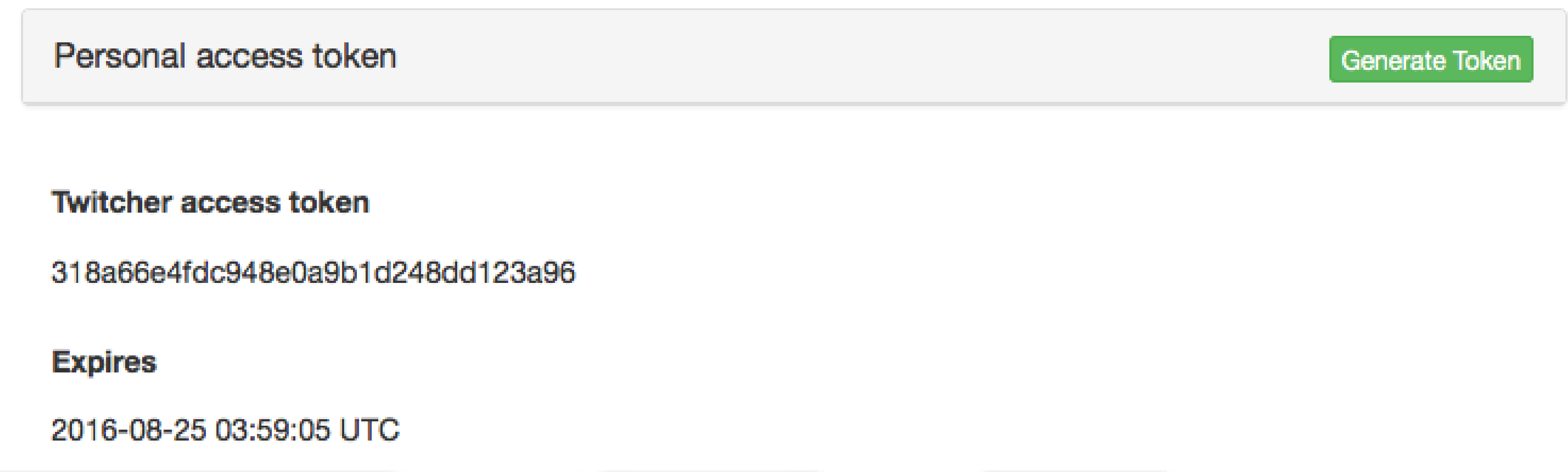
Get Service URL

Registration

Delegation

Service (WPS / WMS)

Use birdy WPS command line client with access token



Personal access token

Twitcher access token

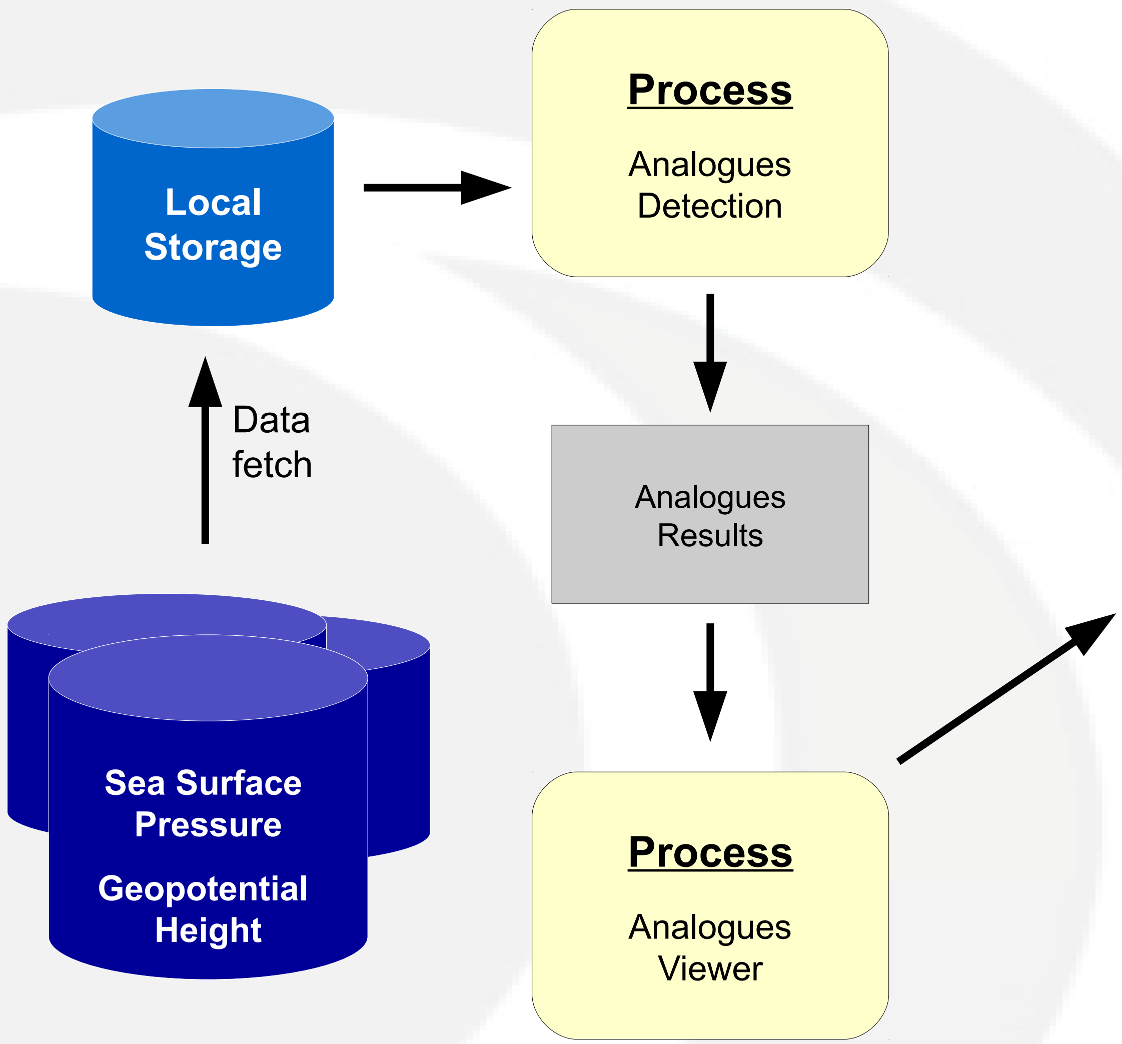
318a66e4fdc948e0a9b1d248dd123a96

Expires

2016-08-25 03:59:05 UTC

Generate access token in Phoenix web app and use it for the birdy WPS command line client to run a protected process.

```
$ conda install -c birdhouse birdhouse-birdy
$ export WPS_SERVICE=https://mywps.demo/ows/proxy/mywps
$ birdy --token 318a66e4fdc948e0a9b1d248dd123a96 \
      cfchecker --dataset http://data.demo/downloads/tasmax.nc
```



Local Storage

Process

Analogues Detection

Data fetch

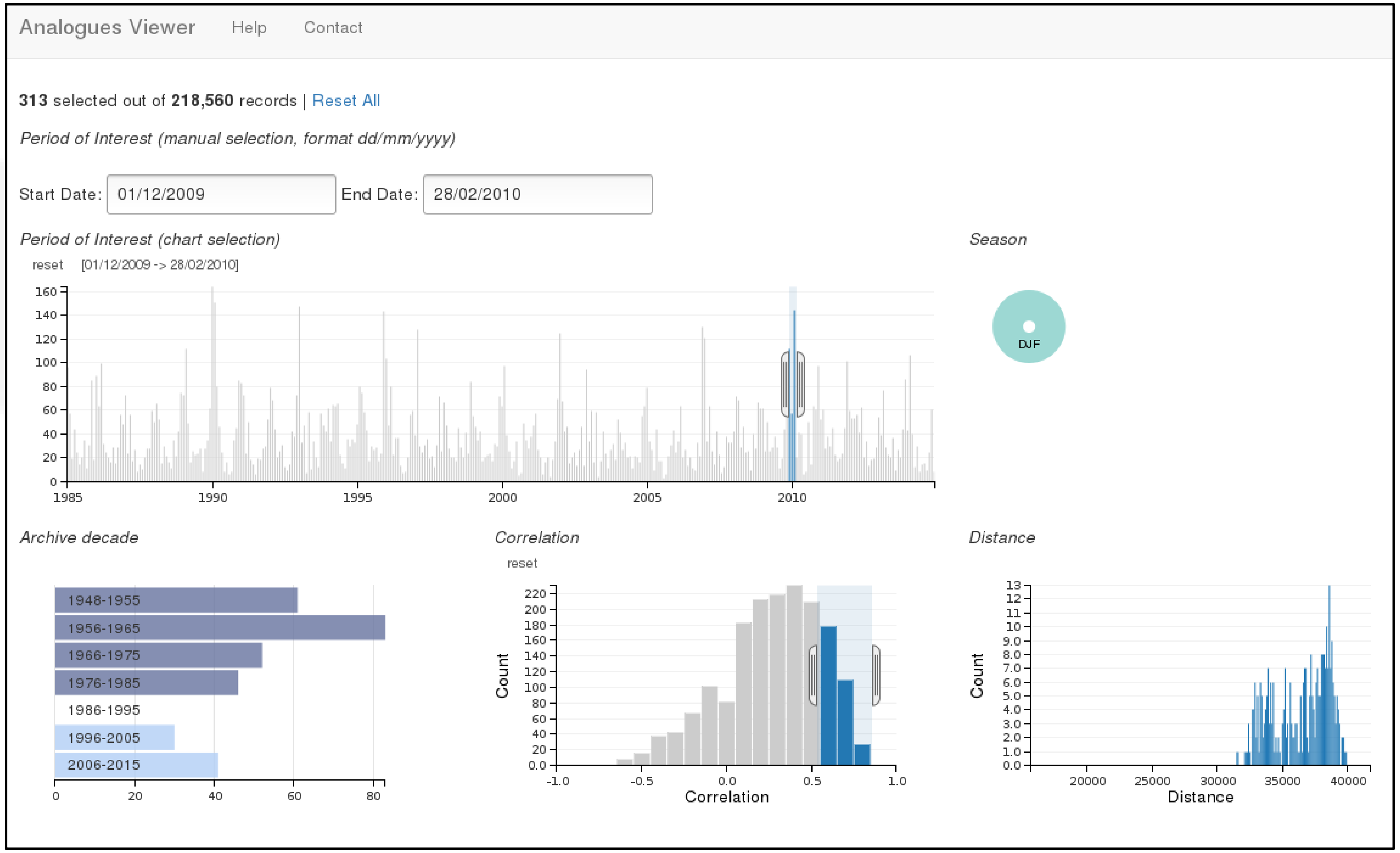
Analogues Results

Sea Surface Pressure

Geopotential Height

Process

Analogues Viewer



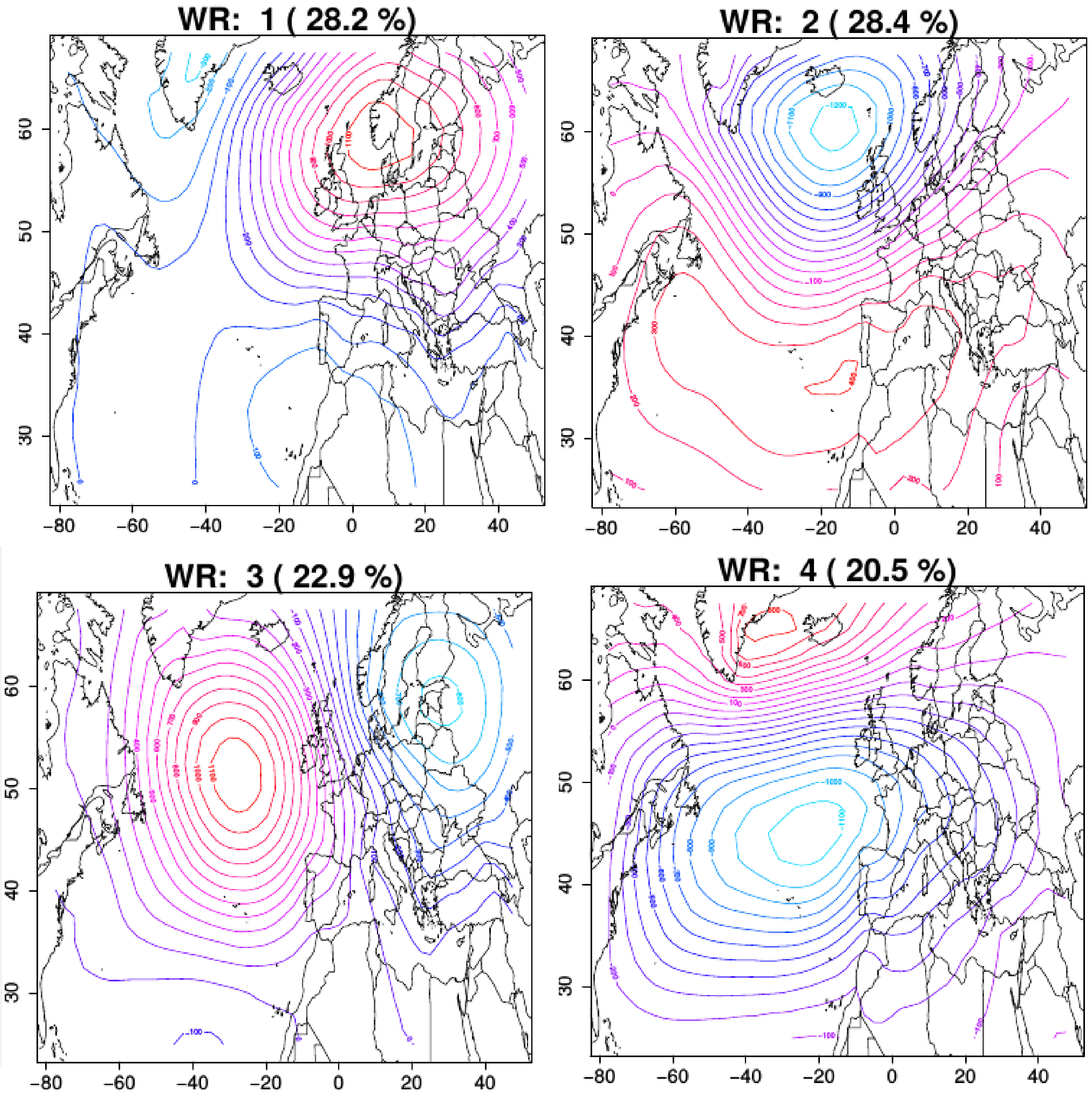
313 selected out of **218,560** records | Reset All

Period of Interest (manual selection, format dd/mm/yyyy)

Start Date: 01/12/2009 End Date: 28/02/2010

Period of Interest (chart selection)

reset [01/12/2009 -> 28/02/2010]


WR: 1 (28.2 %)

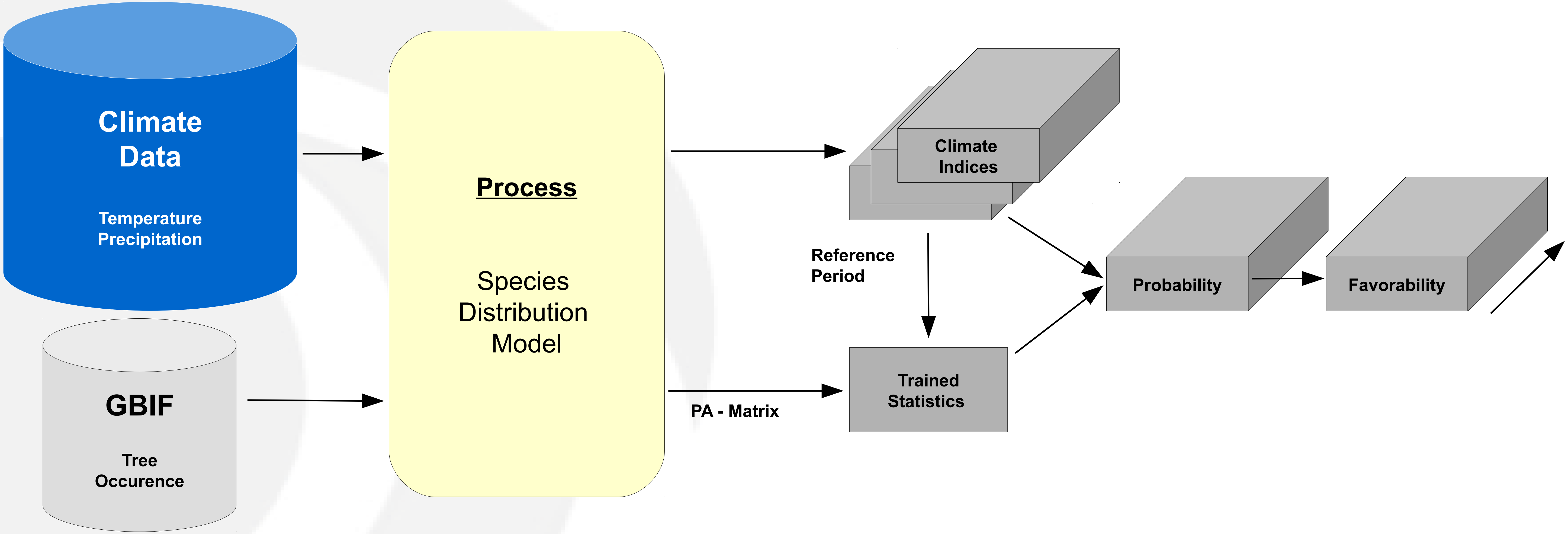
WR: 2 (28.4 %)

WR: 3 (22.9 %)

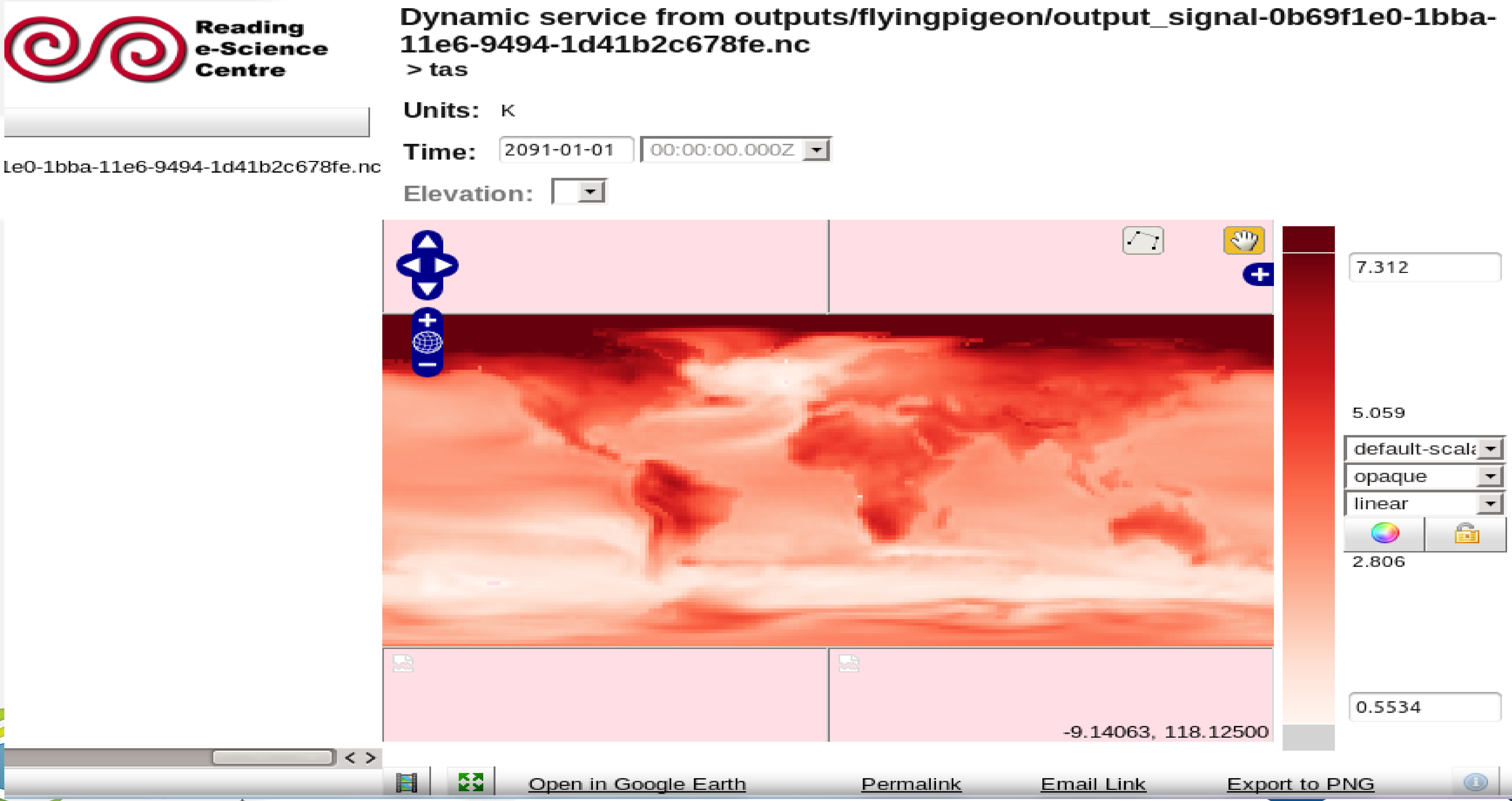
WR: 4 (20.5 %)

Trained Weather regimes Projected on other Dataset

Year	WR 1	WR 2	WR 3	WR 4
...				
2084	35.16	28.57	30.77	5.49
2085	33.33	24.44	36.67	5.56
2086	21.11	28.89	40.00	10.00
2087	37.78	11.11	10.00	41.11
2088	18.68	19.78	37.36	24.17
2089	34.44	44.44	17.78	3.33
...				


Climate Data
Temperature
Precipitation
GBIF
Tree
Occurence
Process
Species
Distribution
Model
Climate
Indices
Reference
Period
PA - Matrix
Trained
Statistics
Probability
Favorability

Web Mapping Server



Reading e-Science Centre

Le0-1bba-11e6-9494-1d41b2c678fe.nc

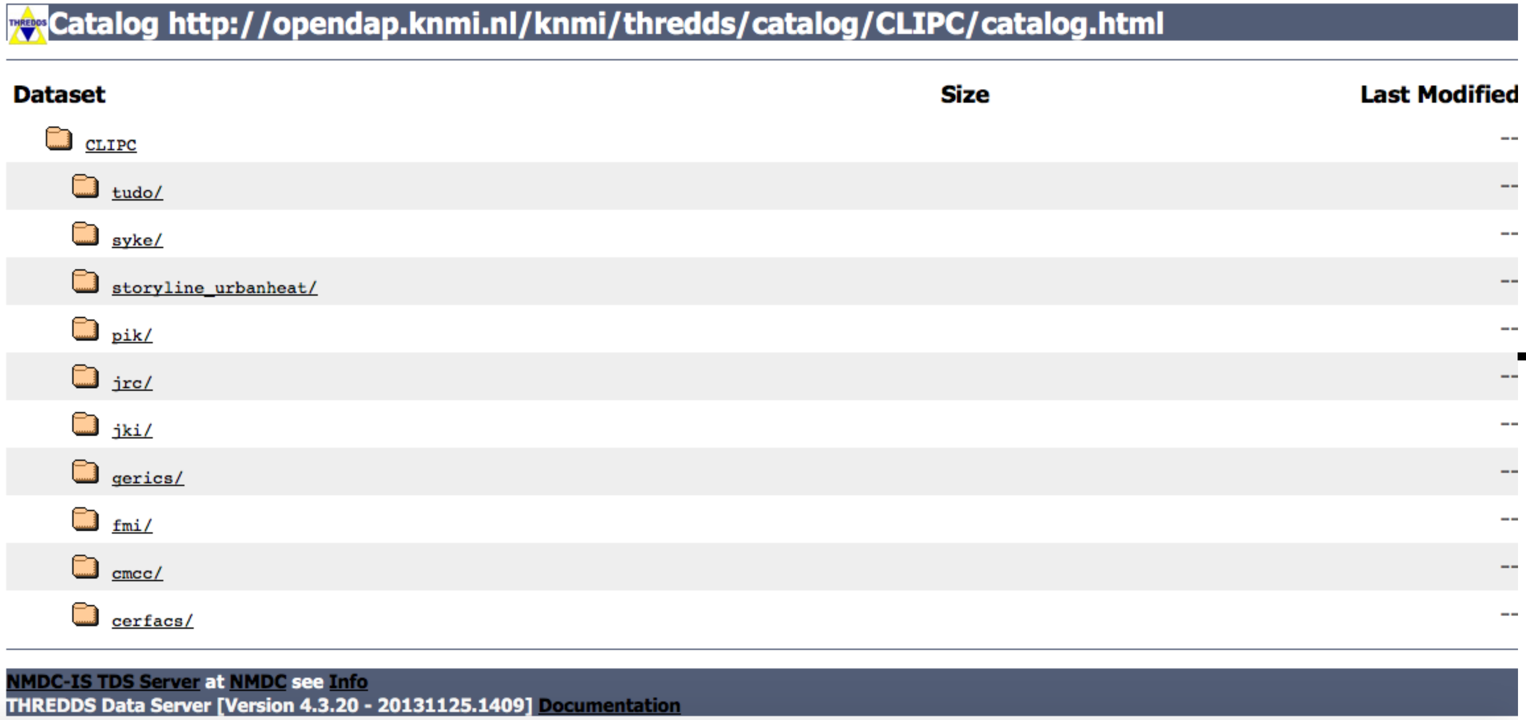
Dynamic service from outputs/flyingpigeon/output_signal-0b69f1e0-1bba-11e6-9494-1d41b2c678fe.nc
> tas

Units: K

Time: 2091-01-01 00:00:00.000Z

Elevation:

Solr Index for Thredds Data Catalogs

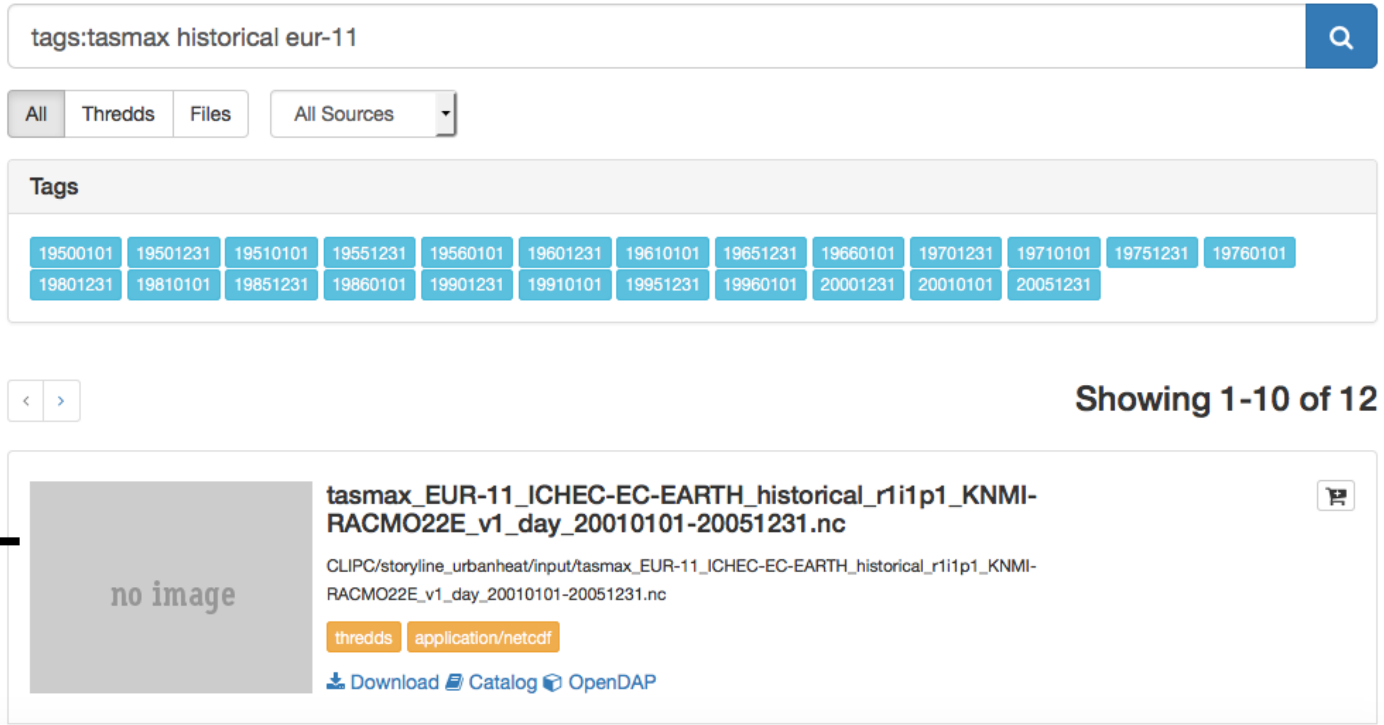


Catalog http://opendap.knmi.nl/knmi/thredds/catalog/CLIPC/catalog.html

Dataset	Size	Last Modified
CLIPC		--
tudo/		--
syke/		--
storyline_urbanheat/		--
pik/		--
jrc/		--
jki/		--
gerics/		--
fmi/		--
cmcc/		--
cerfacs/		--

NMDC-IS TDS Server at NMDC see Info
THREDDS Data Server [Version 4.3.20 - 20131125.1409] Documentation

Run bird-feeder to create Solr search Index for Thredds Data Catalogs



tags:tasmax historical eur-11

Tags

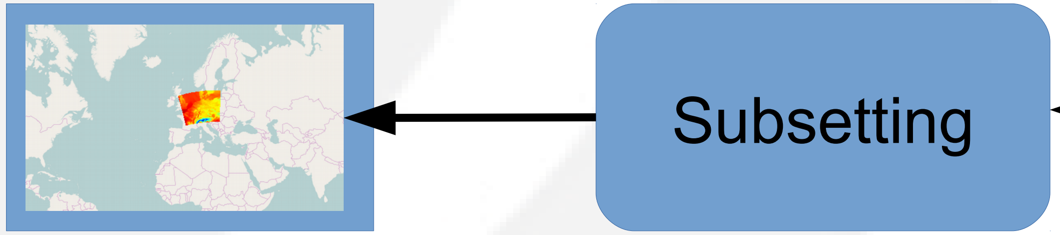
19500101 19501231 19510101 19551231 19560101 19601231 19610101 19651231 19660101 19701231 19710101 19751231 19760101 19801231 19810101 19851231 19860101 19901231 19910101 19951231 19960101 20001231 20010101 20051231

Showing 1-10 of 12

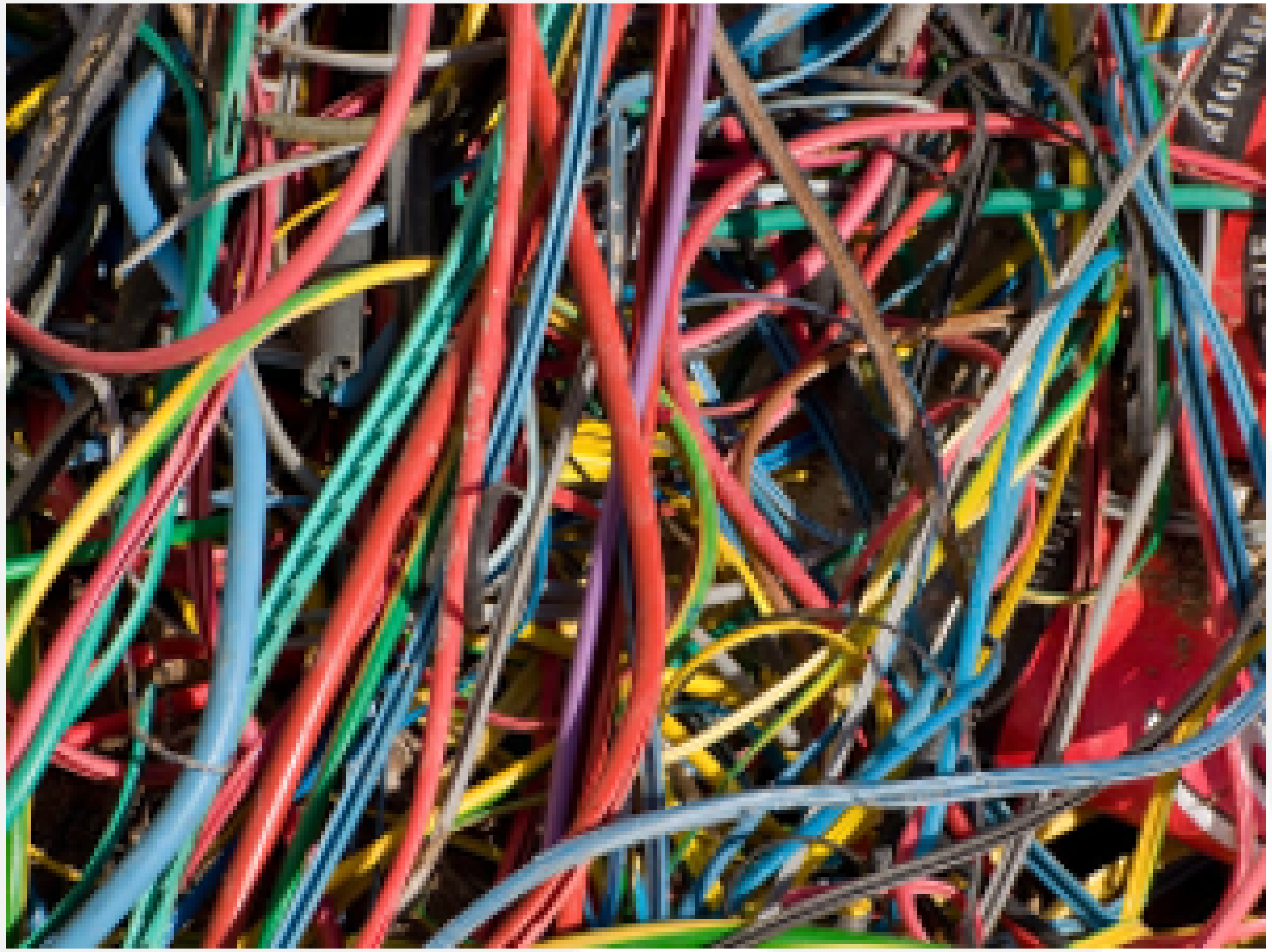
tasmax_EUR-11_ICHEC-EC-EARTH_historical_r1i1p1_KNMI-RACMO22E_v1_day_20010101-20051231.nc

CLIPC/storyline_urbanheat/input/tasmax_EUR-11_ICHEC-EC-EARTH_historical_r1i1p1_KNMI-RACMO22E_v1_day_20010101-20051231.nc

Select data in solr search view and run process



Deployment with conda and buildout



Using conda package manager to setup an environment with all used software components (python, R, matplotlib, PyWPS, ...)

Using buildout to setup PyWPS with all services (supervisor, gunicorn, nginx) and configuration files.

To install a *Bird* just run :

```
$ git clone ...
$ make install
$ make start
```

http://conda.pydata.org/docs/

http://www.buildout.org/en/latest/

http://birdhouse.readthedocs.io/en/latest/installation.html

- https://github.com/bird-house
- http://birdhouse.readthedocs.org/en/latest/

- https://gitter.im/bird-house/birdhouse
- https://lists.dkrz.de/mailman/listinfo/wps
- https://lists.dkrz.de/mailman/listinfo/wps-dev

- DEMO GUI: https://mouflon.dkrz.de

Contact :

ehbrecht[a]dkrz.de
info[a]nilshempelmann.de

Thanks to :

Carmen Alvarez-Castro, Patrick Brockmann, Carsten Ehbrecht, Wolfgang Falk, Nils Hempelmann, Heinz-Dieter Hollweg, Jörg Hoffmann, Nikolay Kadygrov, Stephan Kindermann, Florian Klemme, Nikolay Koldunov, Ben Koziol, Cathy Nangini, Sabine Radanovics, Seckmag, Robert Vautard, Pascal Yiou , …. , et. al.

DKRZ