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

info[a]nilshempelmann.de

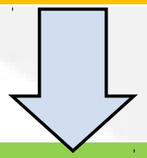
FOSS4G BONN 2016

DKRZ

cnrs

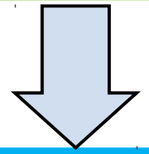
Climate Data volume grows quickly

But on client side:
Limited storage/compute capacities



“download and process at home”

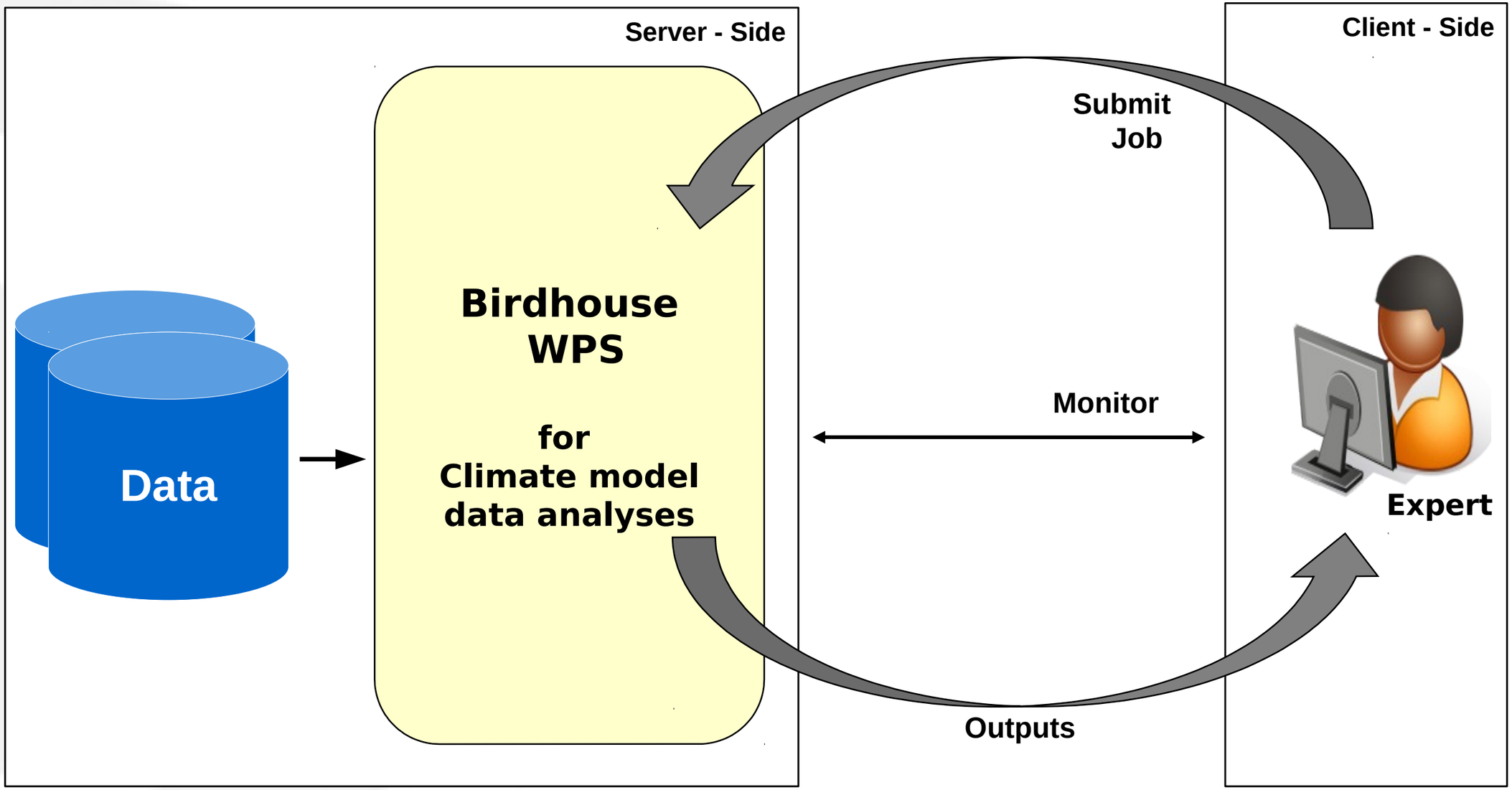
Processing in or close to Data archives



Web Processing Service

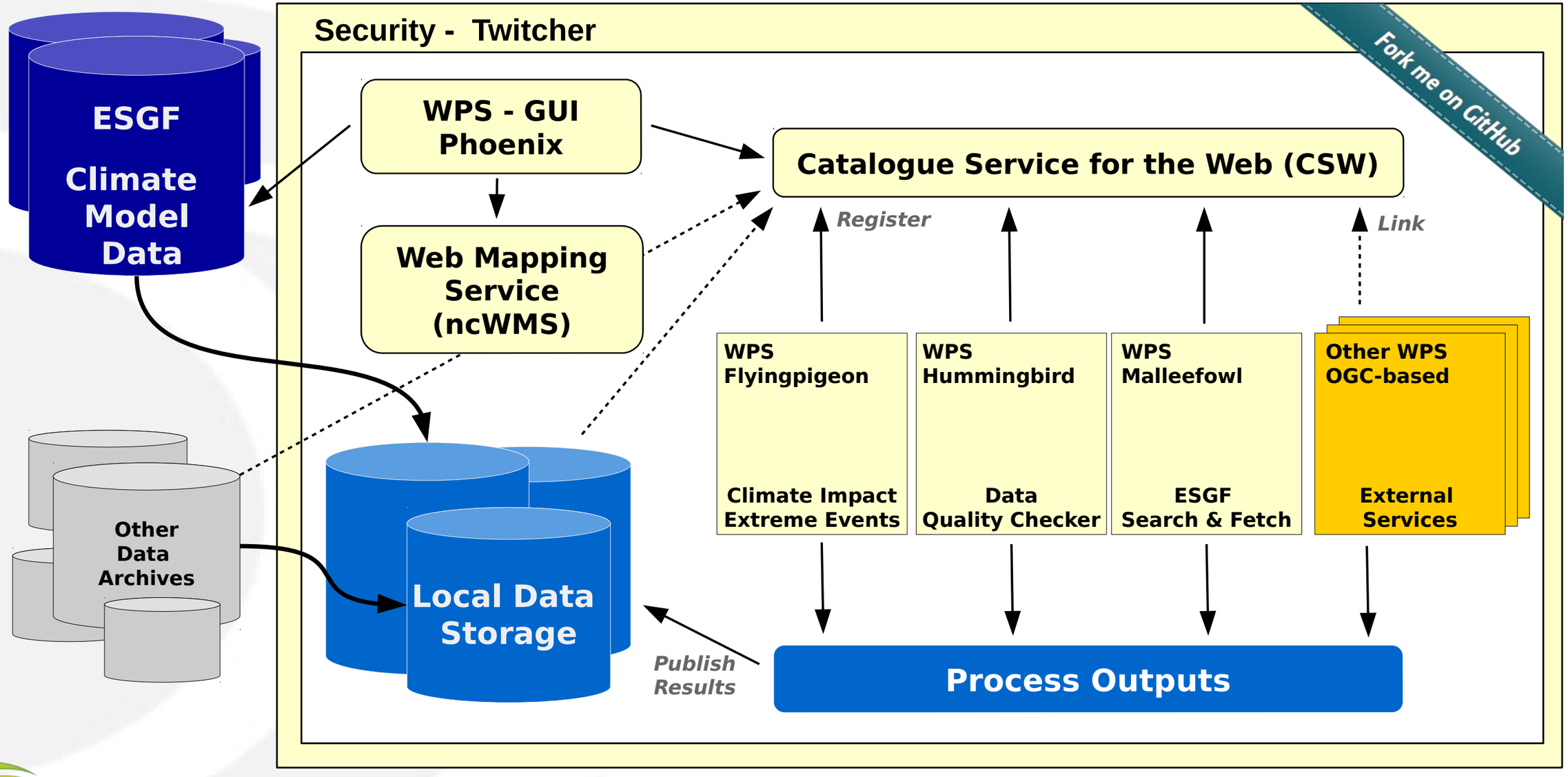
Submit jobs on a Server close to the data

Server-Client Side



Birdhouse - Ecosystem

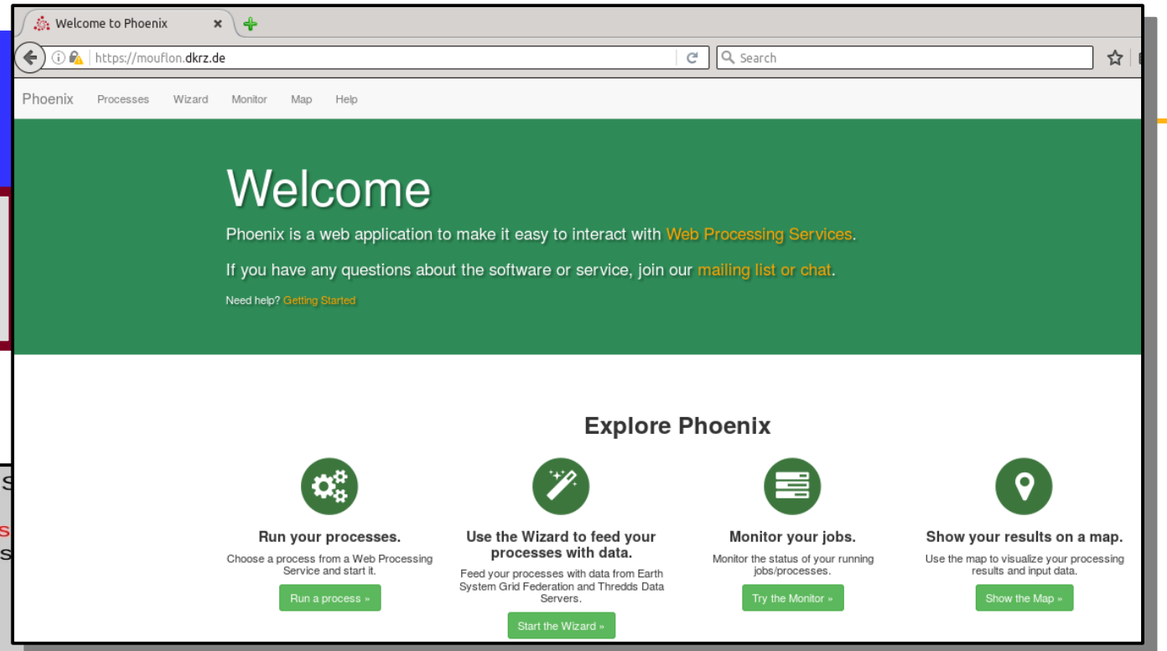
https://github.com/bird-house/



Client Side

Web Browser GUI

Authentication with OAuth or OpenID



Script language Terminal Call

Token authentication

```
[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_t
                  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
```

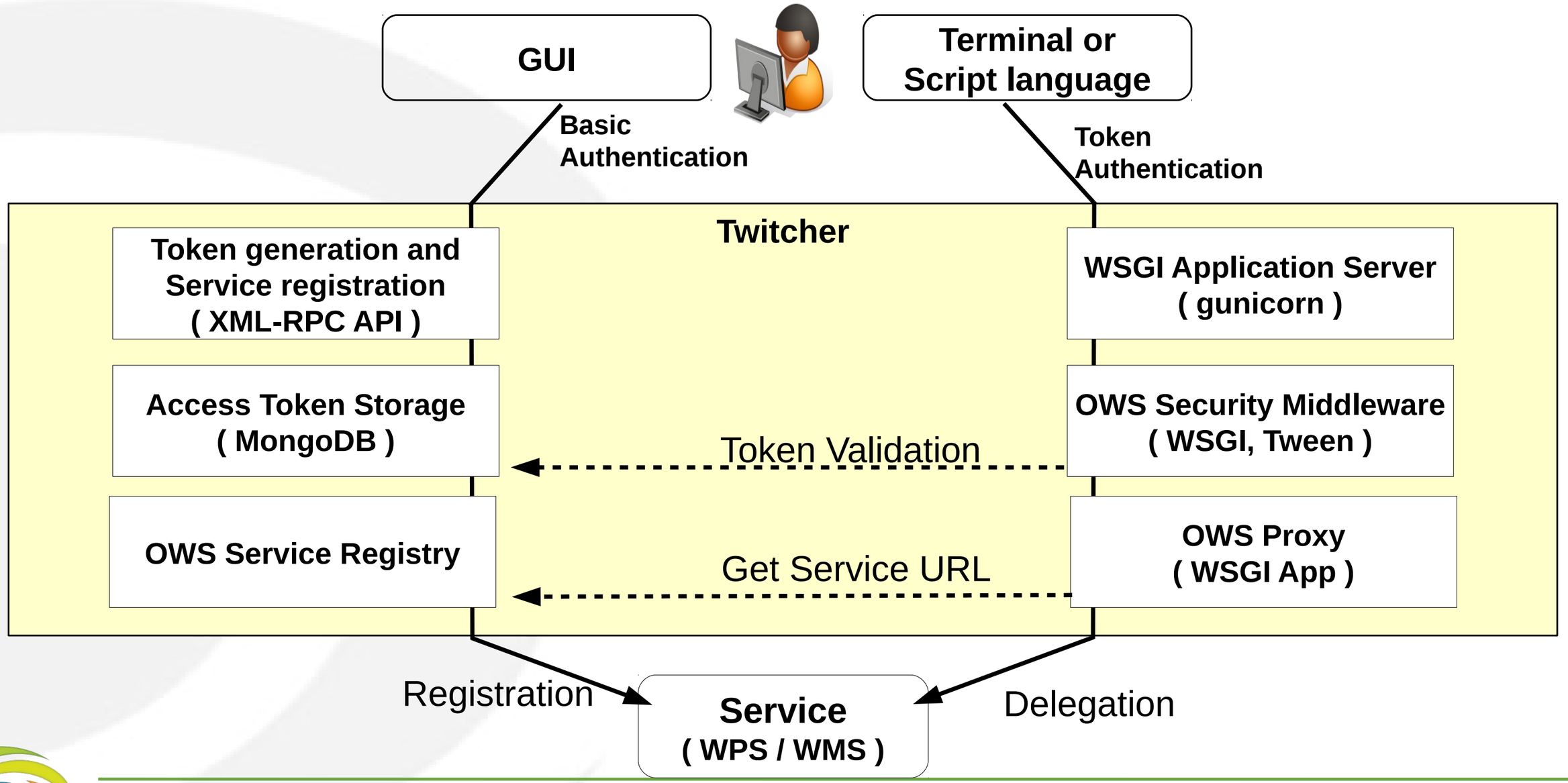
```
from owslib.wps import WebProcessingS

wps = WebProcessingService(url="https
                           verbose=False, s

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
```

info[a]nilshempelmann.de

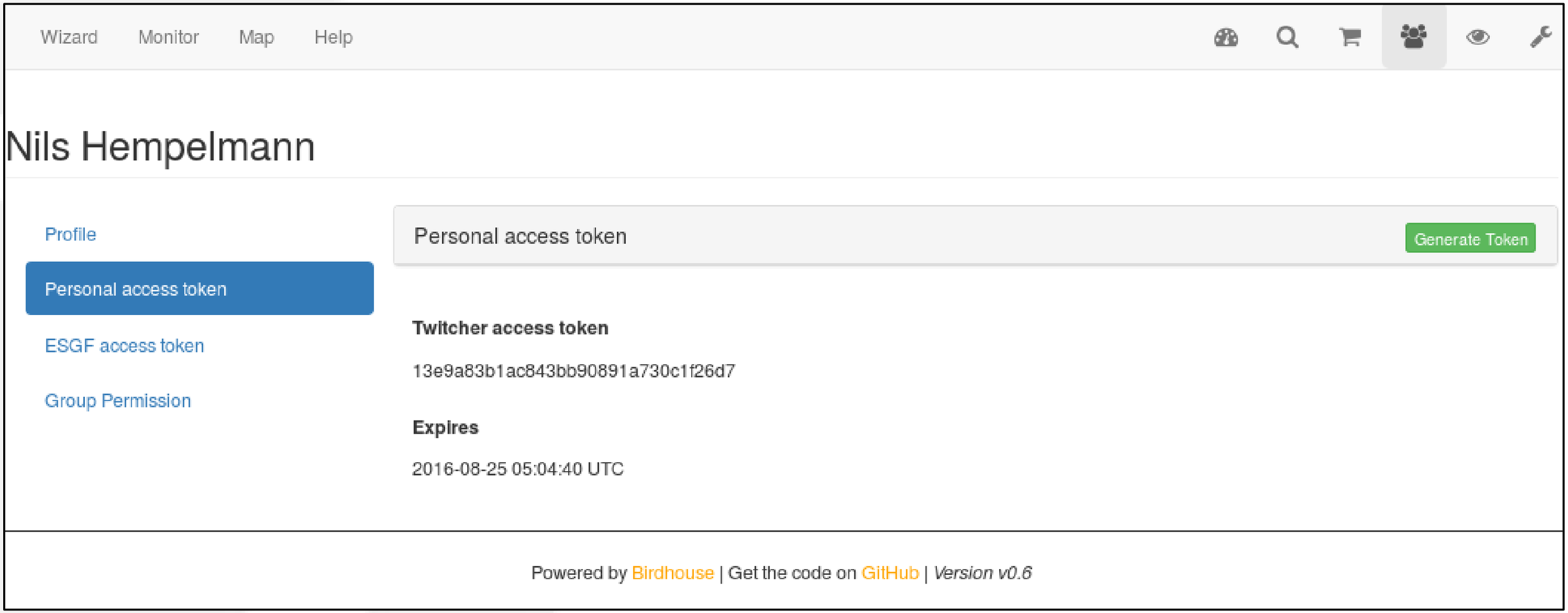
FOSS4G BONN 2016

DKRZ cnrs

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
Service
(WPS / WMS)
Delegation

Security Token



Script language

```
from owslib.wps import WebProcessingService, monitorExecution

wps = WebProcessingService(
                        url="https://mouflon.dkrz.de/ows/proxy/flyingpigeon/db6c1293d0444d919dcc3ce48fa610f7 ", \
                        verify=False,
                        verbose=False, skip_caps=False,
                        )

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)])

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
```

Terminal Call

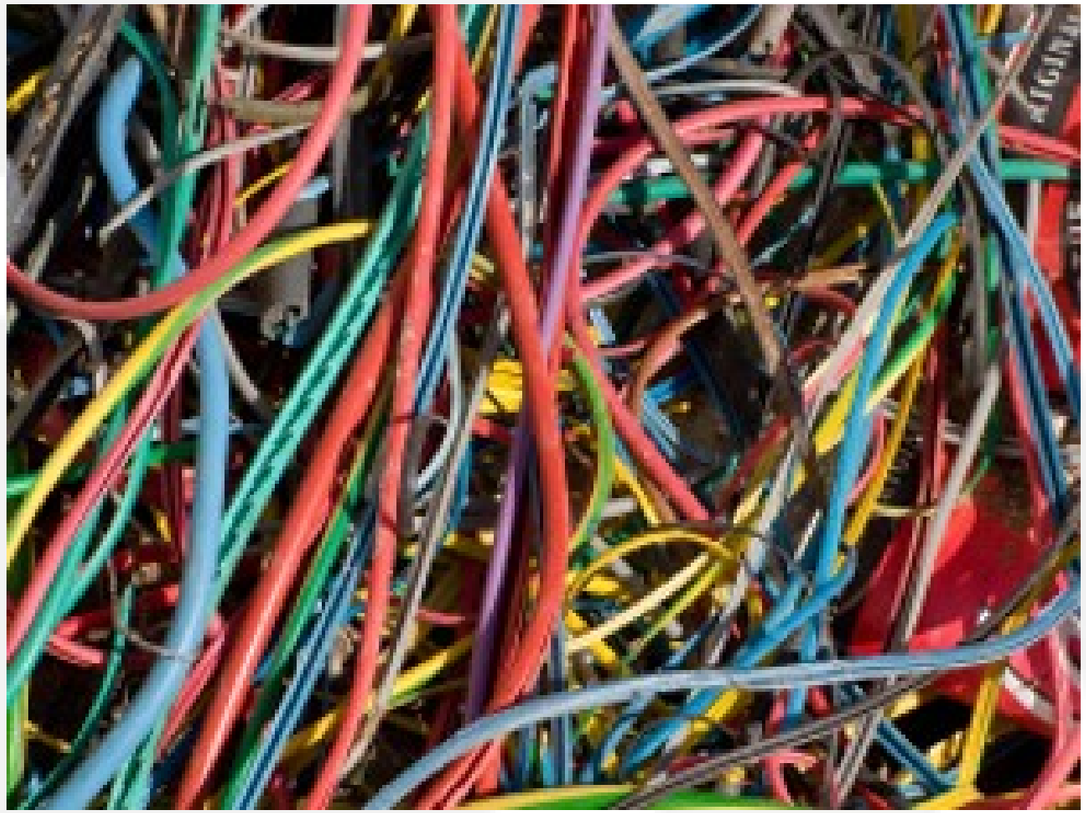
```
[nhempel@lsce3199 ~]$ conda install -c birdhouse birdhouse-birdy

[nhempel@lsce3199 ~]$ birdy -h
usage: birdy [<options>] <command> [<args>]

Flyingpigeon: Processes for climate data, indices and extreme events

optional arguments:
  -h, --help            show this help message and exit
  --debug               enable debug mode
  --token TOKEN, -t TOKEN
                        Token to access the WPS service.

command:
  List of available commands (wps processes)
```

Terminal Call

```
[nhempel@lsce3199 ~]$ export WPS_SERVICE=https://mouflon.dkrz.de/ows/proxy/flyingpigeon

[nhempel@lsce3199 ~]$ birdy –token 0c6d305b0f42452cbdcf31c7ac74f1e1 \
                                  analogs_detection --experiment 'NCEP_slp'

INFO:Execution status: ProcessAccepted
INFO:Execution status: ProcessStarted
INFO:Execution status: ProcessSucceeded

INFO:Output:

INFO:analogs=http://mouflon.dkrz.de/wpsoutputs/flyingpigeon/analogs-08bce60c-6a41-11e6-be7a-8fdf4b12fcf5.txt (text/plain)
INFO:config=http://mouflon.dkrz.de/wpsoutputs/flyingpigeon/config-08bce60c-6a41-11e6-be7a-8fdf4b12fcf5.txt (text/plain)

[nhempel@lsce3199 ~]$
```

http://twitcher.readthedocs.io/en/latest/tutorial.html

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 :

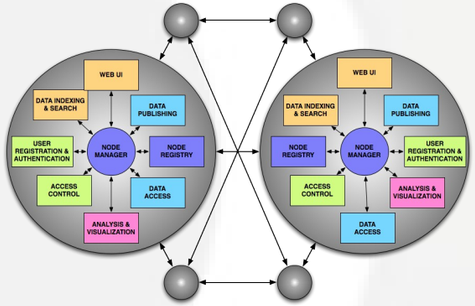
```
$ git clone https://github.com/bird-house/flyingpigeon.git
$ cd flyingpigeon
$ make install
$ make start
```

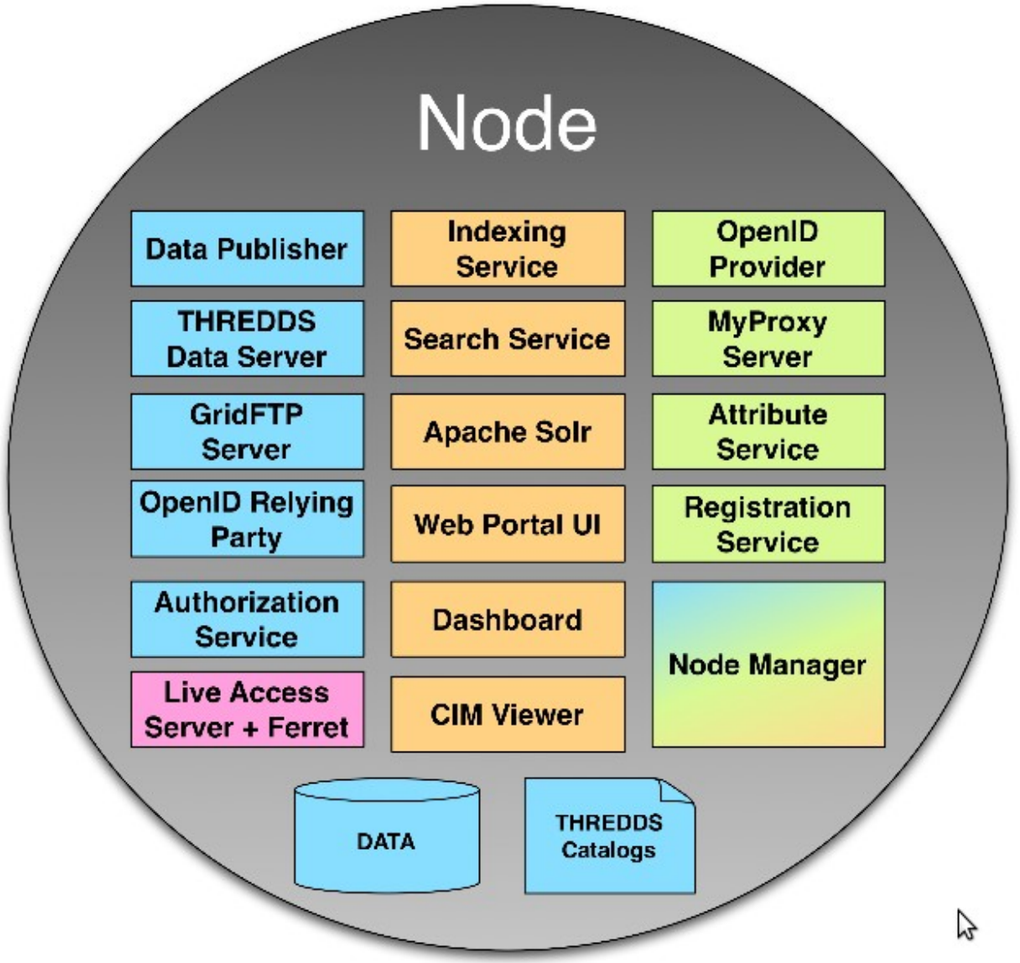
http://conda.pydata.org/docs/

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

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

ESGF Climate Data Archive

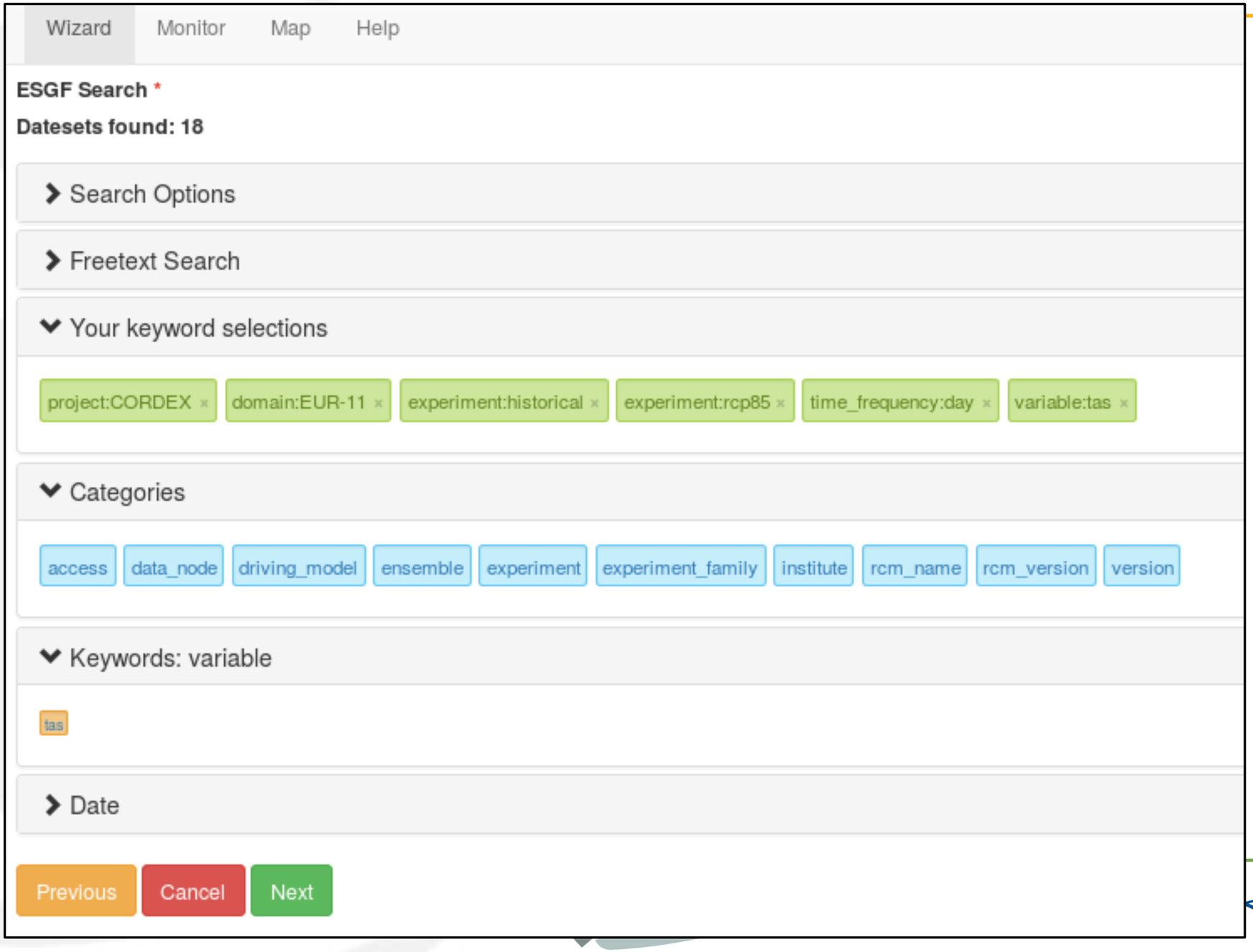




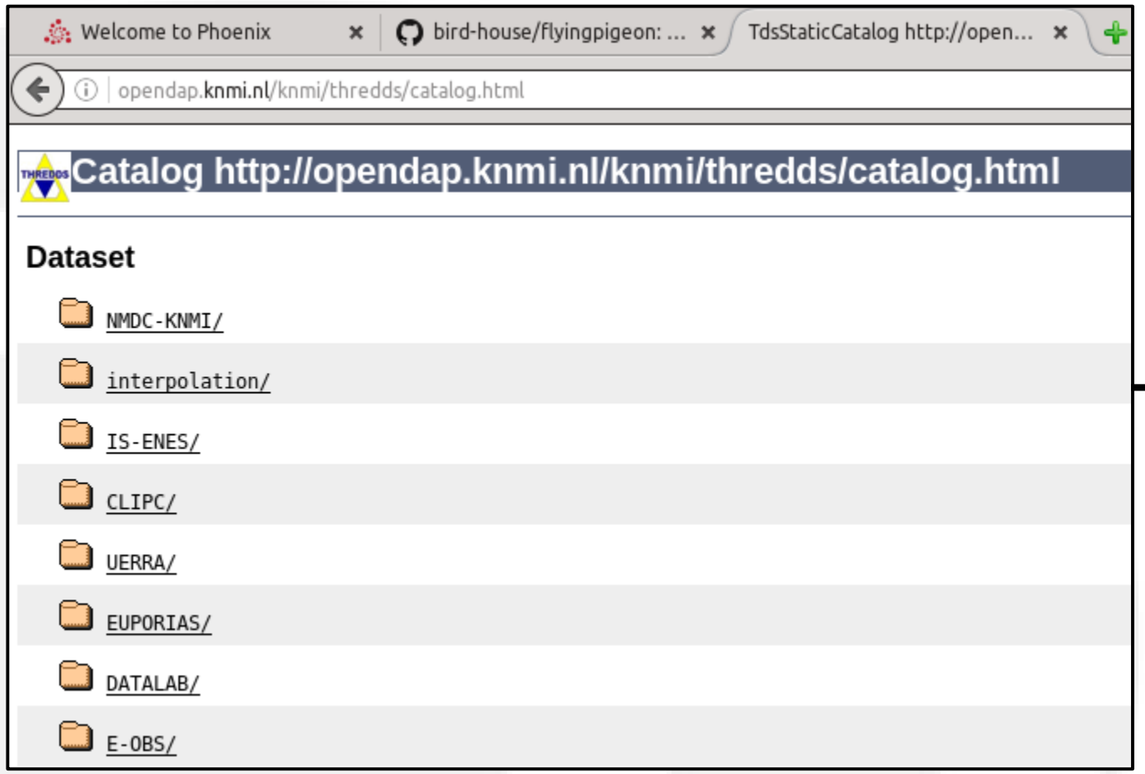
Earth System Grid Federation

https://esgf-data.dkrz.de/

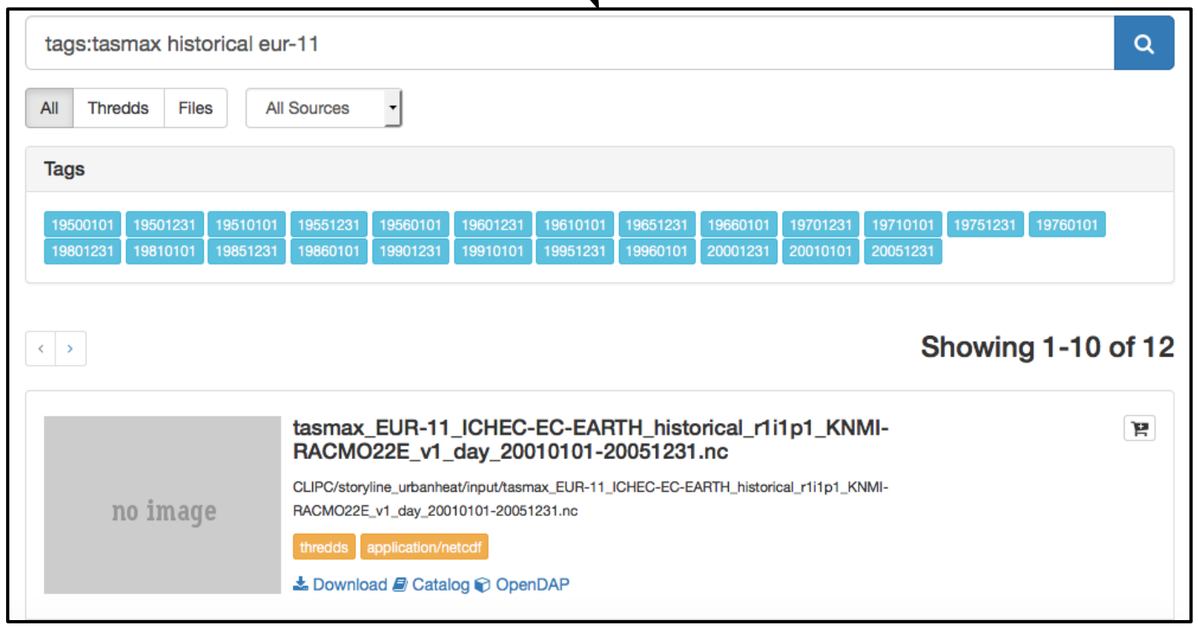
ESGF – search


Wizard
Monitor
Map
Help
ESGF Search *
Datesets found: 18
Search Options
Freetext Search
Your keyword selections
project:CORDEX ×
domain:EUR-11 ×
experiment:historical ×
experiment:rcp85 ×
time_frequency:day ×
variable:tas ×
Categories
access
data_node
driving_model
ensemble
experiment
experiment_family
institute
rcm_name
rcm_version
version
Keywords: variable
tas
Date
Previous
Cancel
Next

Solr Index for Data (**bird-feeder**)



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



Hummingbird – quality checks for netCDF Data (technical)

NetCDF Metadata - Retrieve Metadata of NetCDF File

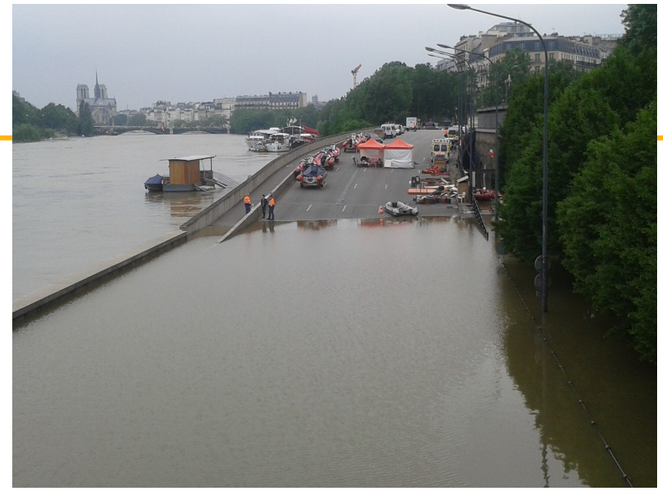
CF Checker by NCAS Computational Modelling Services (NCAS-CMS) - The NetCDF Climate Forcast Conventions compliance checker. This process allows you to run the compliance checker to check that the contents of a NetCDF file comply with the Climate and Forecasts (CF) Metadata Convention. The CF-checker was developed at the Hadley Centre for Climate Prediction and Research, UK Met Office by Rosalyn Hatcher. This work was supported by PRISM (PRogramme for Integrated Earth System Modelling). Development and maintenance for the CF-checker has now been taken over by the NCAS Computational Modelling Services (NCAS-CMS). If you have suggestions for improvement then please contact Rosalyn Hatcher at NCAS-CMS (r.s.hatcher@reading.ac.uk).

CF Checker by DKRZ - The NetCDF Climate Forcast Conventions compliance checker by DKRZ. This process allows you to run the compliance checker to check that the contents of a NetCDF file comply with the Climate and Forecasts (CF) Metadata Convention. The CF Conformance checker applies to conventions 1.4 -1.7draft. Development and maintenance for the CF-checker is done by the German Climate Computing Centre (DKRZ). If you have suggestions for improvement then please contact Heinz-Dieter Hollweg at DKRZ (hollweg@dkrz.de).

Quality Assurance Checker by DKRZ - The Quality Assurance checker QA-DKRZ checks conformance of meta-data of climate simulations given in NetCDF format with conventions and rules of climate model projects. At present, checking of CF Conventions, CMIP5, and CORDEX is supported. Development and maintenance for the QA checker is done by the German Climate Computing Centre (DKRZ). If you have suggestions for improvement then please contact Heinz-Dieter Hollweg at DKRZ (hollweg@dkrz.de).

IOOS Compliance Checker - The IOOS Compliance Checker is a Python tool to check local/remote datasets against a variety of compliance standards. Each compliance standard is executed by a Check Suite, which functions similar to a Python standard Unit Test. A Check Suite runs one or more checks against a dataset, returning a list of Results which are then aggregated into a summary. Development and maintenance for the compliance checker is done by the Integrated Ocean Observing System (IOOS).

Flyingpigeon - - Climate Impact and extreme events



Subset continents - Returns only the selected polygon for each input dataset
Subset countries - Returns only the selected polygon for each input dataset
Subset Points - Extract Timeseries for specified coordinates from gridded datasets

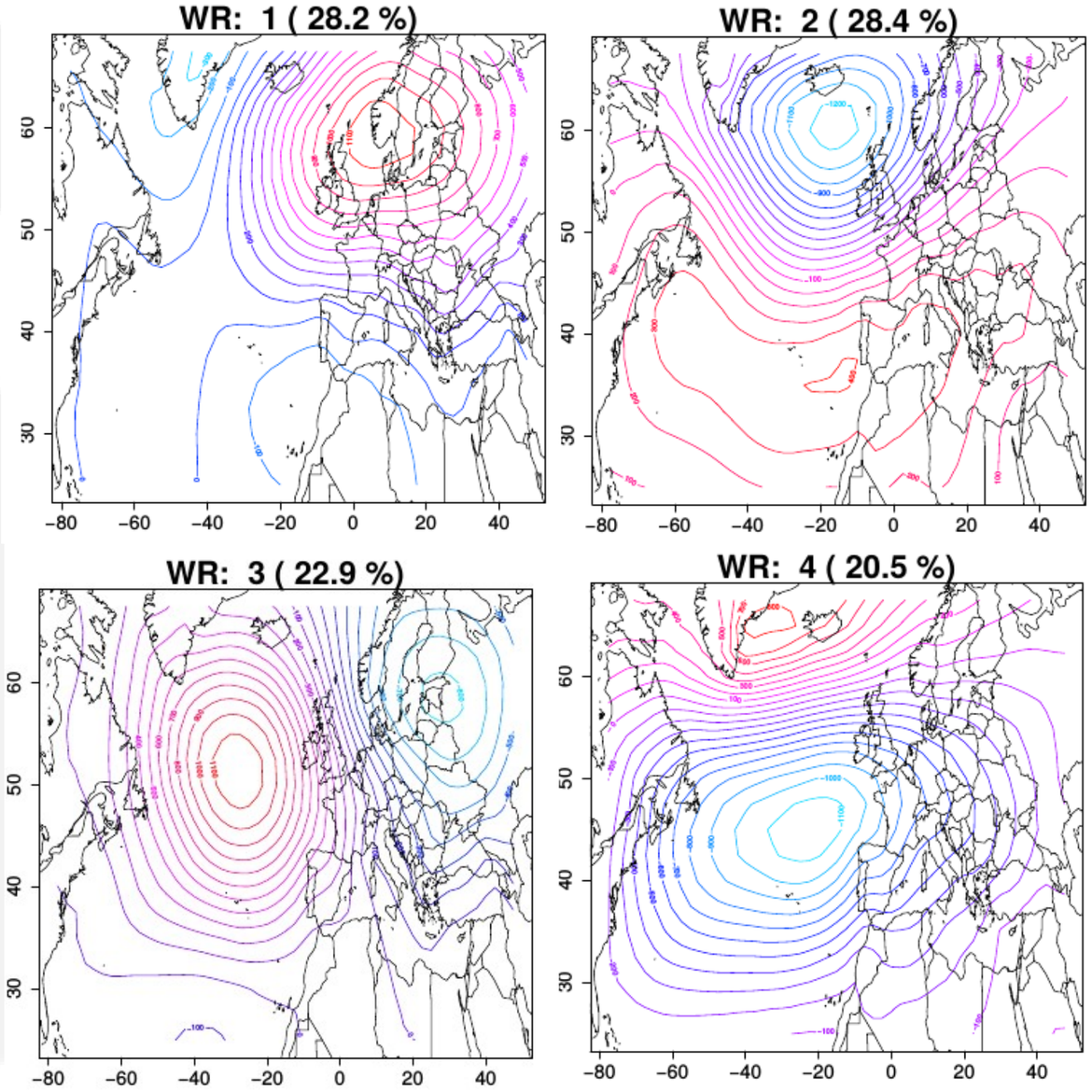
Climate indices -- Simple - Climate indices based on one single input variable.
Climate indices -- Percentile - Climate indices based on one single input variable and the percentile of a reference period.

Weather Regimes -- Reanalyses data - Weather Regimes based on pressure patterns, fetching selected Realayses Datasets
Weather Regimes -- Climate model data - Weather Regimes based on pressure patterns, fetching selected Realayses Datasets
Weather Regimes -- Projection of Weather Regimes - Weather Regimes detection based on trained reference statistics
Analogs -- Detection - Search for days with analog pressure pattern
Analogs -- Viewer - Visualisation of text output of analogue process

Segetal Flora - Species biodiversity of segetal flora. Imput files: variable:tas , domain: EUR-11 or EUR-44
SDM -- GBIF search - Species distribution model for tree species based on GBIF presens/absence data and climate indices
SDM -- csv table - Species distribution model for tree species based on GBIF presens/absence data and climate indices
Timeseries plots - Plots of the filesmeans over time. Spagetti and uncertainty plot
Download Resources - This process downloads resources to the local file system of the birdhouse compute provider

More under development ;-)

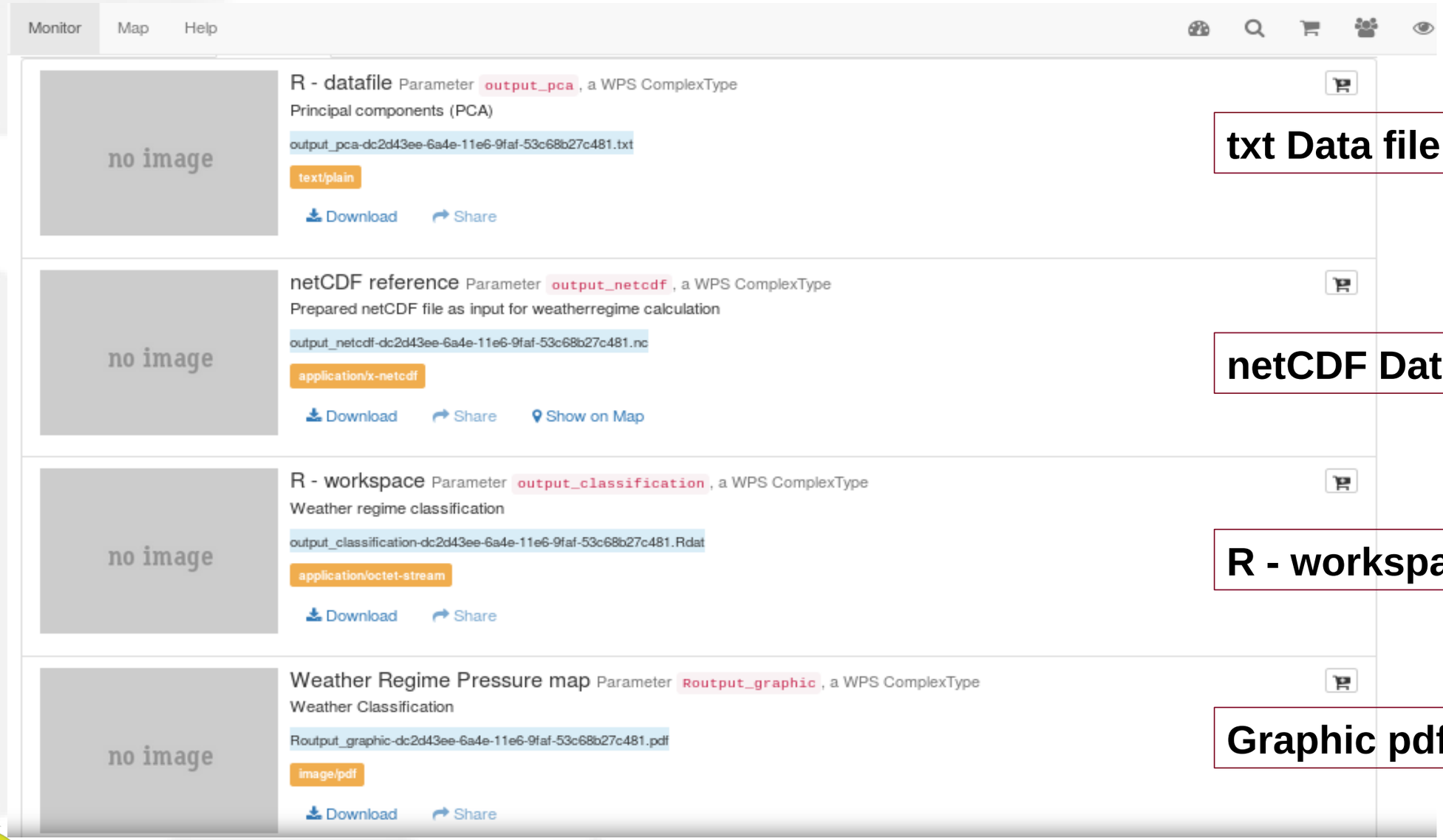
Weather regimes



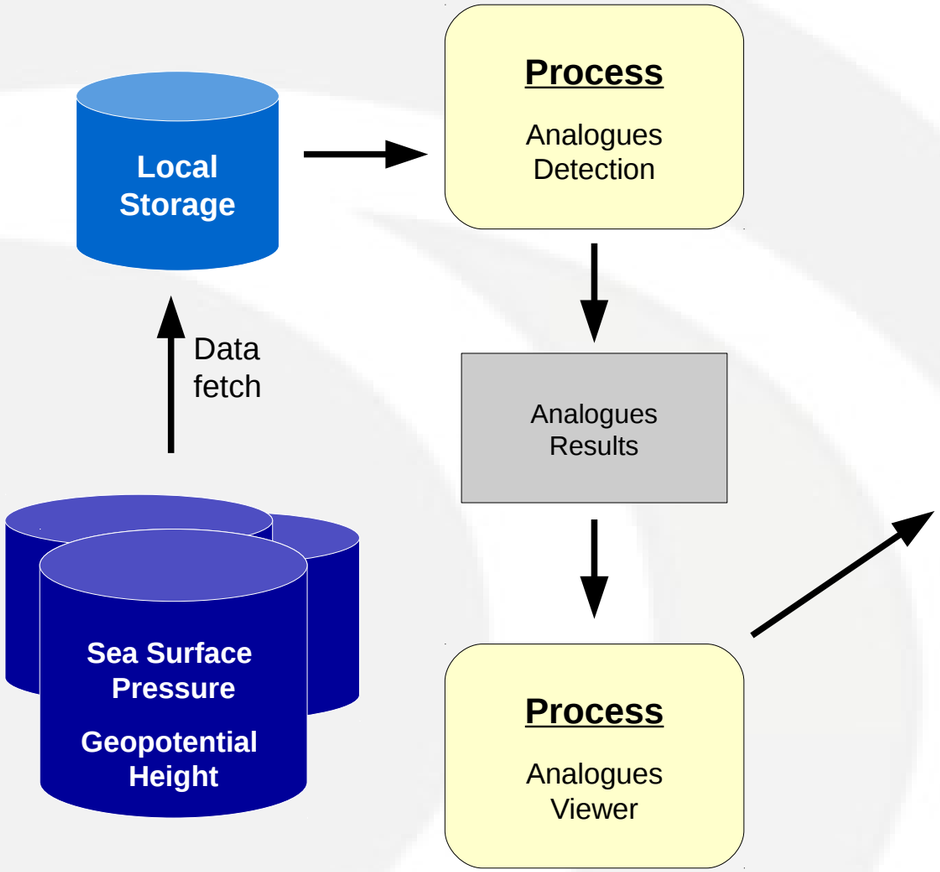
Trained Weather regimes Projected on other Dataset

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
...				

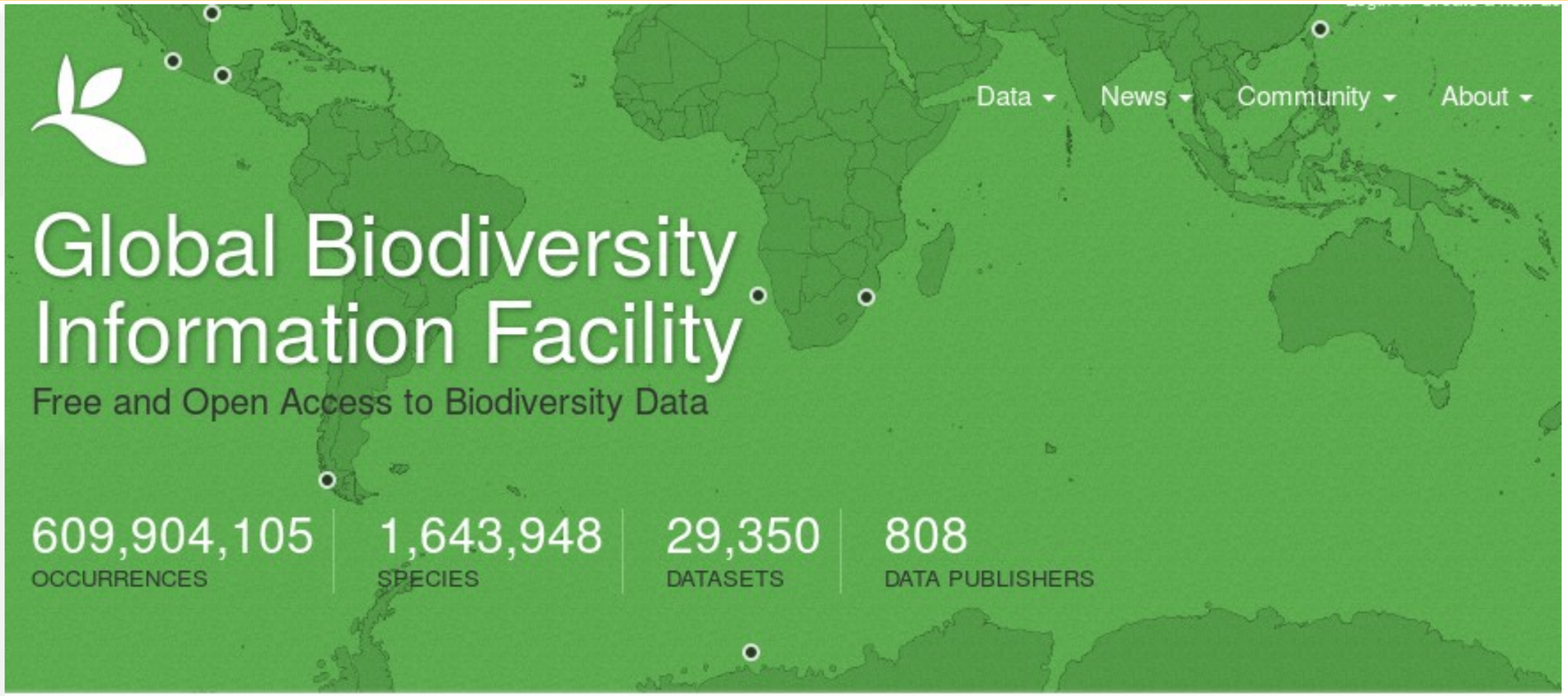
info[a]nilshempelmann.de


Monitor
Map
Help
no image
R - datafile Parameter output_pca, a WPS ComplexType
Principal components (PCA)
output_pca-dc2d43ee-6a4e-11e6-9faf-53c68b27c481.txt
text/plain
Download
Share
txt Data file
no image
netCDF reference Parameter output_netcdf, a WPS ComplexType
Prepared netCDF file as input for weatherregime calculation
output_netcdf-dc2d43ee-6a4e-11e6-9faf-53c68b27c481.nc
application/x-netcdf
Download
Share
Show on Map
netCDF Data file
no image
R - workspace Parameter output_classification, a WPS ComplexType
Weather regime classification
output_classification-dc2d43ee-6a4e-11e6-9faf-53c68b27c481.Rdat
application/octet-stream
Download
Share
R - workspace
no image
Weather Regime Pressure map Parameter Routput_graphic, a WPS ComplexType
Weather Classification
Routput_graphic-dc2d43ee-6a4e-11e6-9faf-53c68b27c481.pdf
image/pdf
Download
Share
Graphic pdf

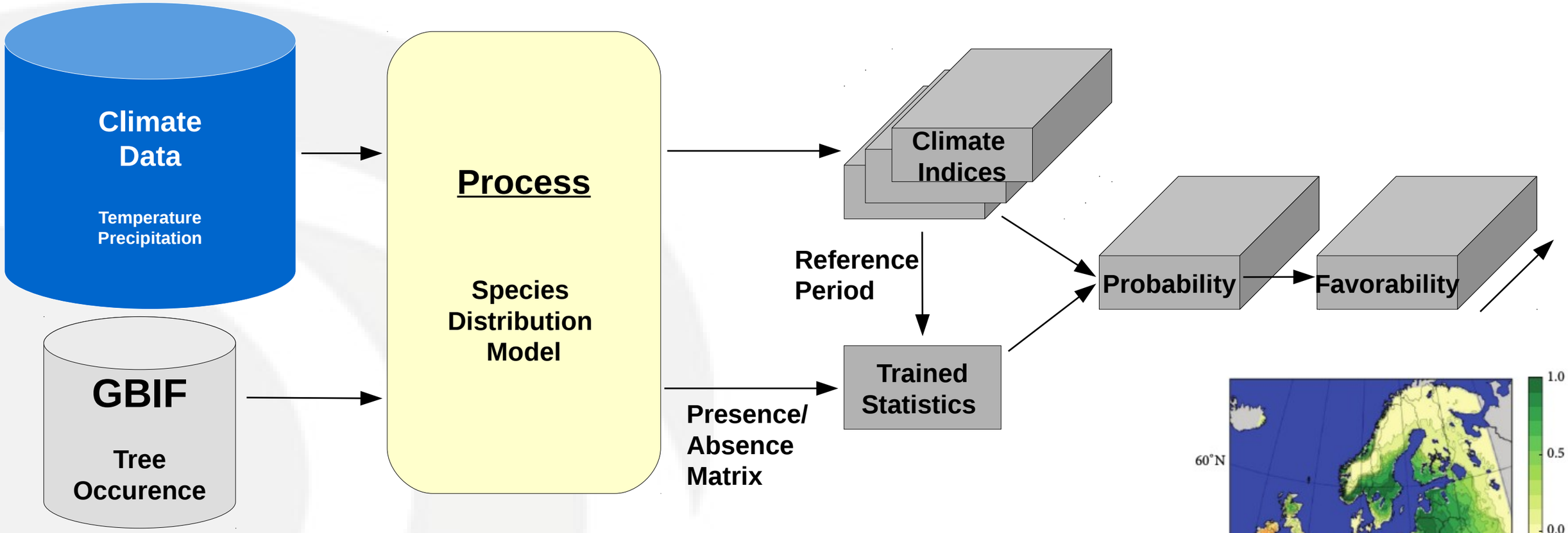
Analogues of atmospheric Circulation



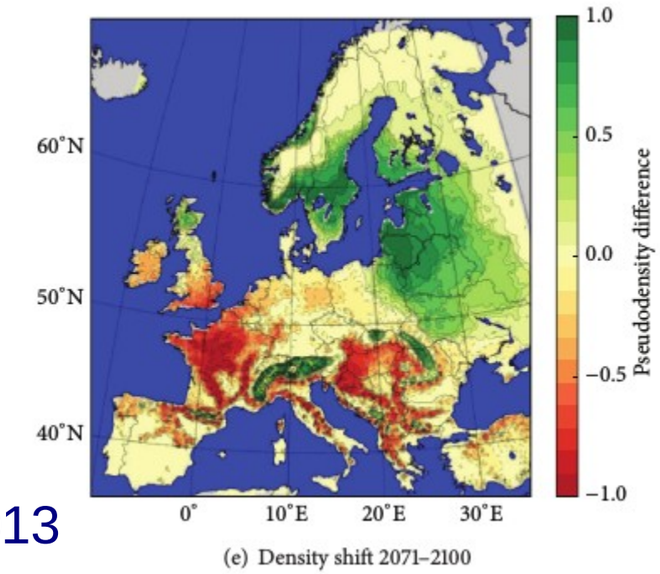
Non-Climate Data



Tree Species distribution model



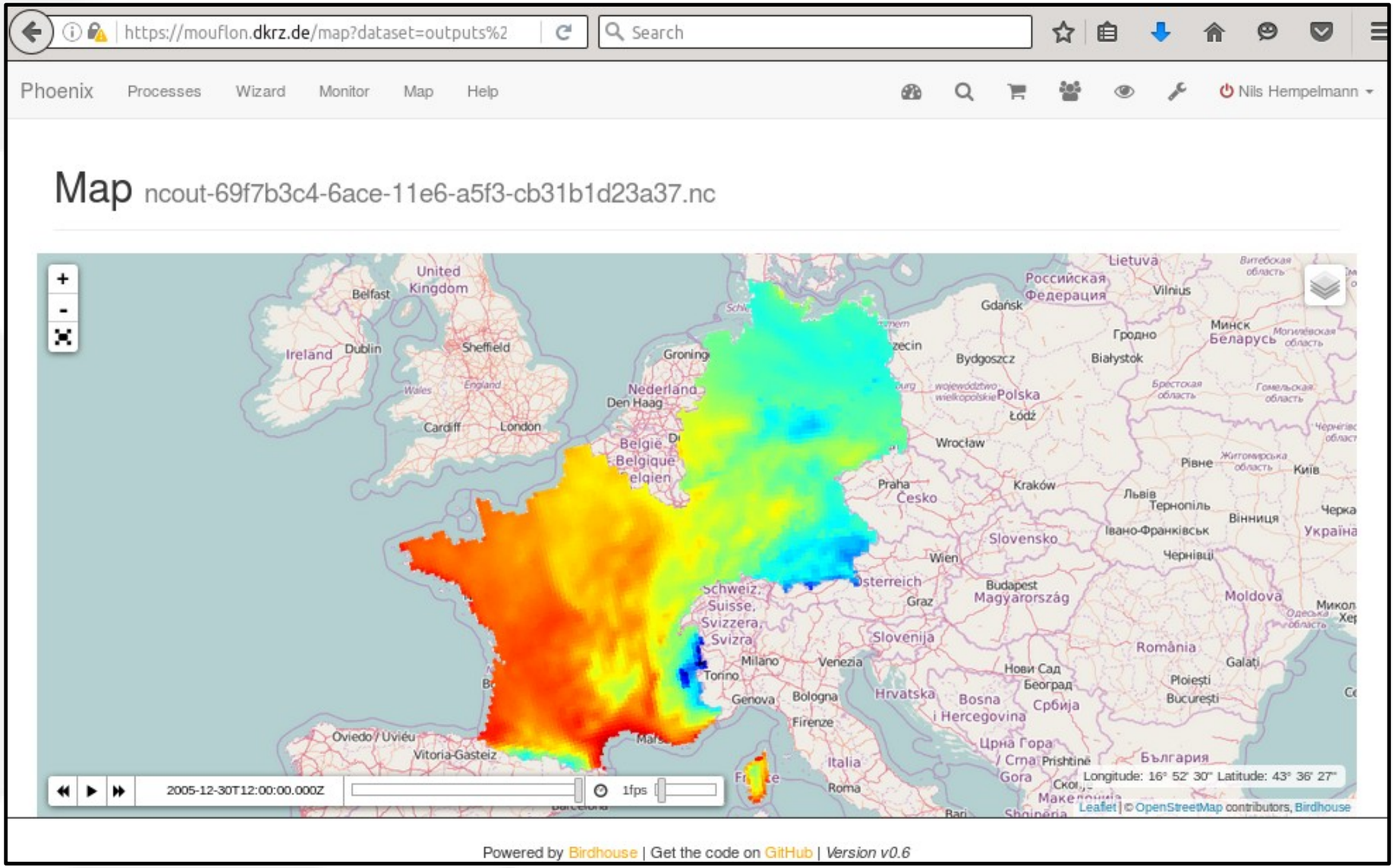
http://www.gbif.org/



(e) Density shift 2071–2100

Falk et al. 2013

Web Mapping Server



- 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, Katharina Berger, 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.