DKRZ ESGF related infrastructure and CMIP6 services

Overview

- DKRZ infrastructure and ESGF
  - Technical infrastructure
  - Data management workflow
- CMIP6 services
  - CMIP data pool
  - CMIP data preparation / ingest / quality assurance
  - Processing, birdhouse WPS ecosystem
  - Long term archival and data citation

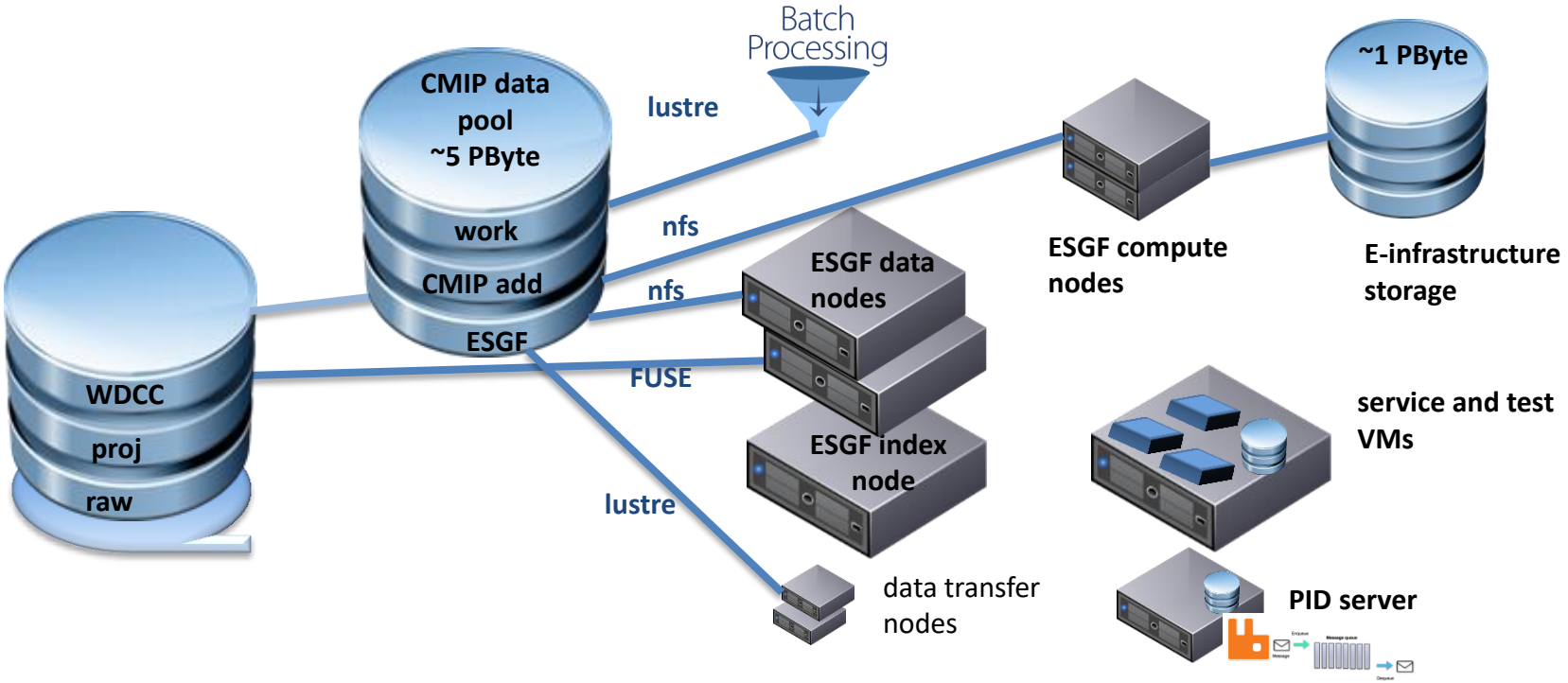
DKRZ technical infrastructure and ESGF

HPSS tape
- 190 Pbyte capacity

Lustre file system
- 54 PByte

„Mistral“ HPC
- 3.6 Pflops
- ~100.000 cores

Compute/ storage cluster
- VM servers, database servers
- Openstack cloud storage

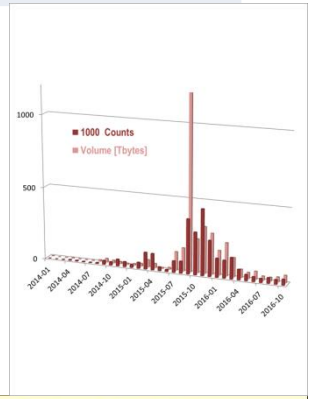


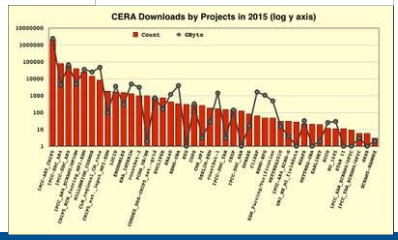
CMIP5 download stats ESGF and WDCC

Months (from .. to)	ESGF – disk (MPI/M- CMIP5 +CORDEX)	ESGF – tape (CMIP5)	ESGF	WDCC – tape CMIP5 replica
2016-10..08	~25 TB/month	~25 TB/month	~50 TB/month	~40 TB/Month
2016-07..01	~25 TB/month	~10 TB/month	~35 TB/month	~200 TB/Month
2015-12..06	0	0	0	~500 TB/Month
2015-05..01	~28 TB/month	0	~28 TB/month	~25 TB/Month

- Substantial CMIP5 replica tape access during ESGF breakdown (1.2 PB/month peak)

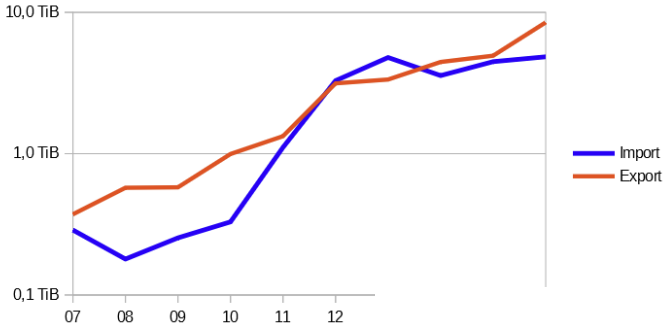
- Since breakdown:
  - All replicas accessible via ESGF/ WDCC (HPSS) data node
  - only CMIP5 MPI-M disk copy republished, other disk replicas not republished to ESGF (local pool)





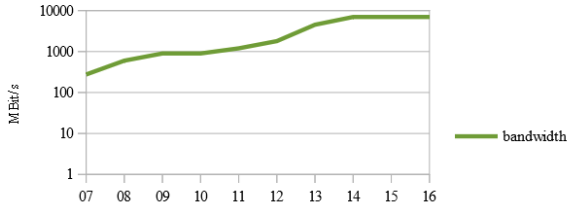
→ Data hub: Pbyte(s)/year In/Output

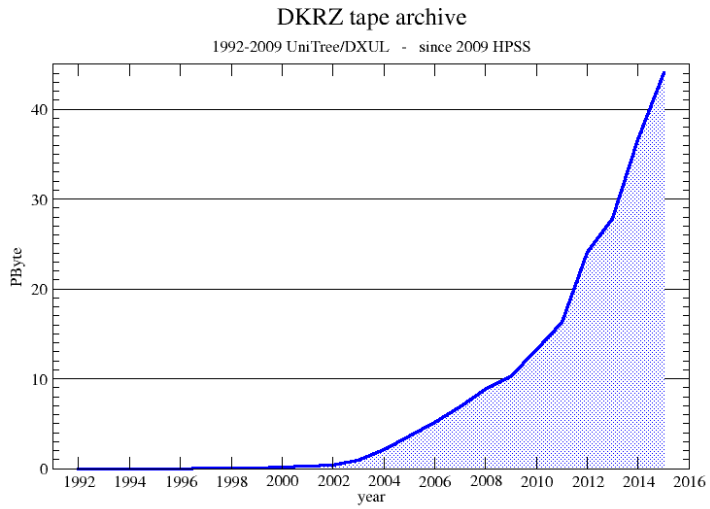
	per year		per month		per day		rate		increase	
Year	**Import**	**Export**	**Import**	**Export**	**Import**	**Export**	**Import**	**Export**	**Import**	**Export**
07	105,1 TiB	135,9 TiB	8,8 TiB	11,3 TiB	0,3 TiB	0,4 TiB	0,0 gbps	0,0 gbps		
08	65,6 TiB	209,6 TiB	5,5 TiB	17,5 TiB	0,2 TiB	0,6 TiB	0,0 gbps	0,1 gbps	0,6	1,5
09	92,6 TiB	211,0 TiB	7,7 TiB	17,6 TiB	0,3 TiB	0,6 TiB	0,0 gbps	0,1 gbps	1,4	1,0
10	120,2 TiB	363,2 TiB	10,0 TiB	30,3 TiB	0,3 TiB	1,0 TiB	0,0 gbps	0,1 gbps	1,3	1,7
11	404,8 TiB	485,2 TiB	33,7 TiB	40,4 TiB	1,1 TiB	1,3 TiB	0,1 gbps	0,1 gbps	3,4	1,3
12	1199,9 TiB	1148,7 TiB	100,0 TiB	95,7 TiB	3,3 TiB	3,1 TiB	0,3 gbps	0,3 gbps	3,0	2,4
13	1748,5 TiB	1221,9 TiB	145,7 TiB	101,8 TiB	4,8 TiB	3,3 TiB	0,5 gbps	0,3 gbps	1,5	1,1
14	1302,8 TiB	1626,5 TiB	108,6 TiB	135,5 TiB	3,6 TiB	4,5 TiB	0,4 gbps	0,5 gbps	0,7	1,3
15	1635,0 TiB	1801,0 TiB	136,3 TiB	150,1 TiB	4,5 TiB	4,9 TiB	0,5 gbps	0,5 gbps	0,9	1,5
16	1769,0 TiB	3100,0 TiB	147,4 TiB	258,3 TiB	4,8 TiB	8,5 TiB	0,5 gbps	0,9 gbps	1,4	1,9
									1,6	1,5



→ HPSS / ESGF data node integration increased replica visibility

→ Demand for data near processing increases

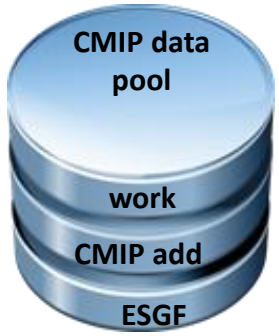




DKRZ CMIP data pool

5 PByte Lustre pool
National coordination board for defining priorities and partitioning

- ~1 PByte german CMIP6 contributions (first estimate)
- ~2 PByte CMIP6 replicas (**content defined by european replication strategy)**
- ~600 TByte CMIP5 data
- ~400 TByte data for analysis and evaluation
- ~ 1 PByte, data management cache
- Final shareholder approval not yet provided



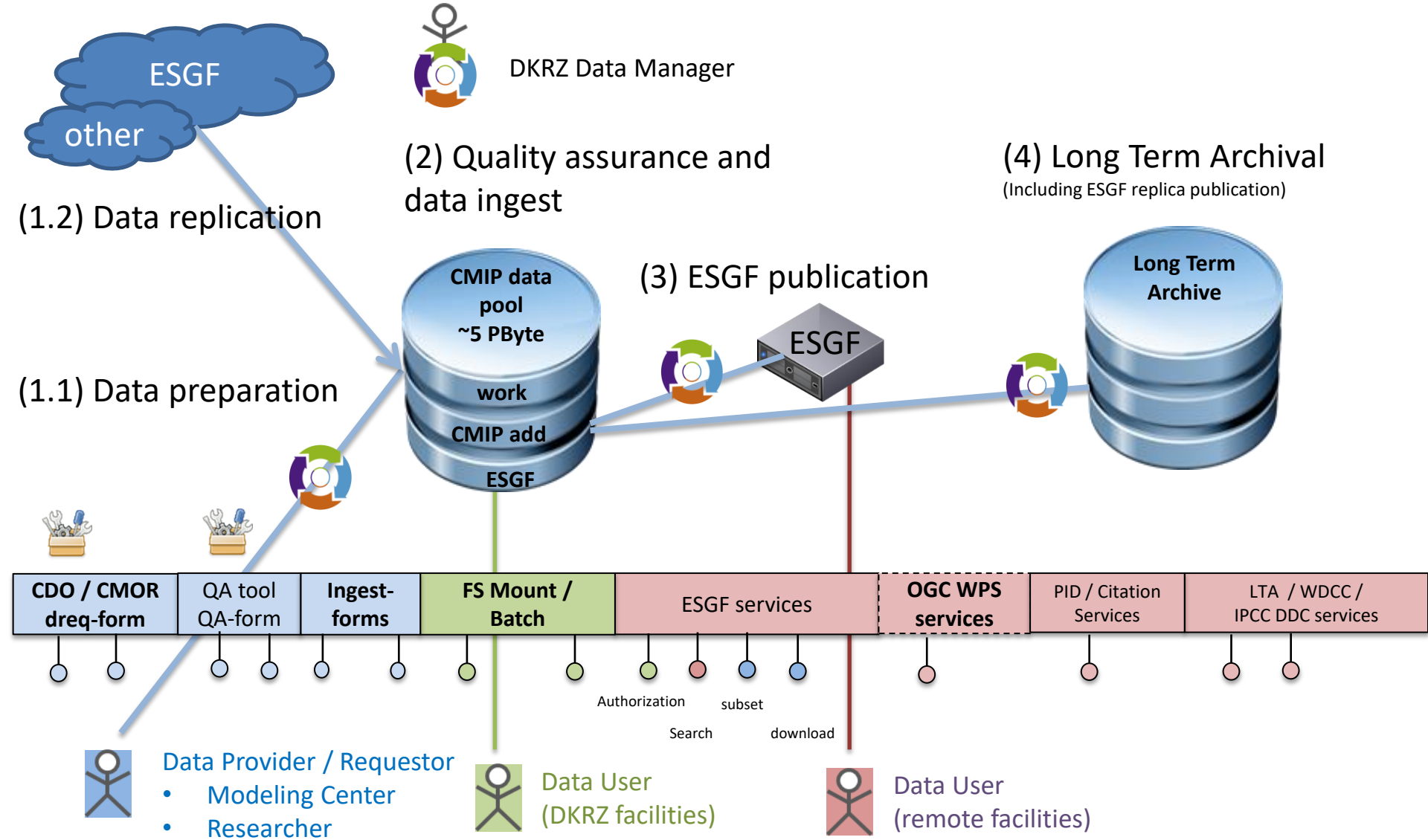
Data Users

Data Near Compute Users

50 PByte Long term Archive

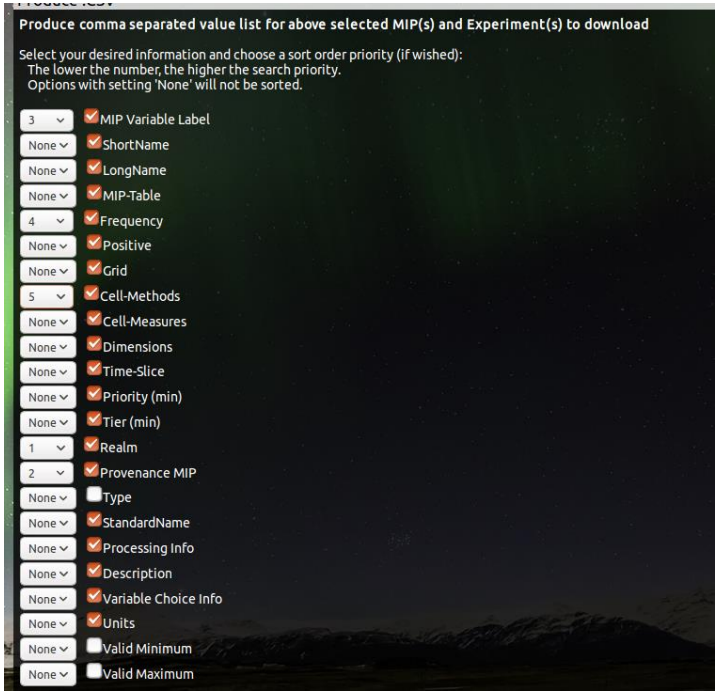
- 20 PByte till 2018 (10 Pbyte real)
- → growing to 100 Pbyte till 2020 (50 Pbyte real)

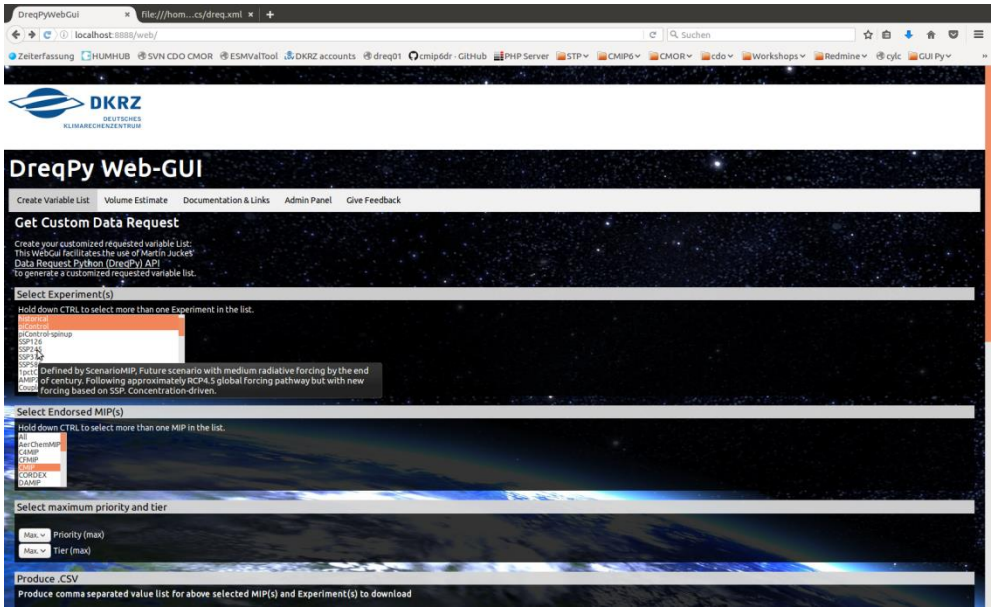
ESGF / CMIP data management workflow and services

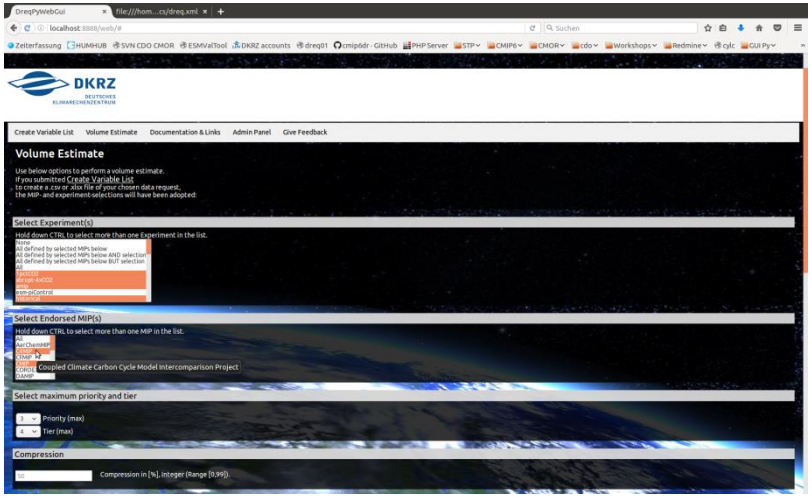


National modeling group support: data request

WebGUI for the generation of specific data requests based on the DreqPy python API (provided by Martin Juckes)

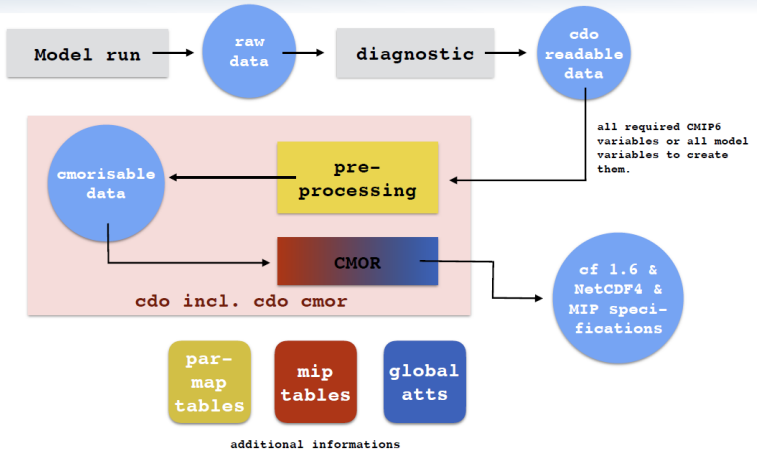






1) Data preparation, cdo/cmor

- Integration of cdo and cmor to support „cmor-compliant“ data generation
- Modification of cdo to support „CMIP-conformance“




```
Cdo cmor,Amon,vars=tas,info=efile,mfile,ufile,ginfo=grid_file ifile
```

```
mip-tables:
cmip6-cmor-tables
with
CMIP6_Amon.json
CMIP6_Omon.json
CMIP6_LImon.json
CMIP6_SImon.json
...
```

```
parameter-tables:
MPIESM1_CMIP5.txt

&parameter code=167 \
name=temp2 \
out_name=tas  \
positive="" \
units="K" \
cell_method=m \
standard_name=\
air_temperature \
delete=0/
```

```
#experiment info
PROJECT_ID=
PRODUCT=
COMMENT=
HISTORY=
EXPERIMENT_ID=
MEMBER=
FORCING=
REQ_TIME_UNITS=
BRANCH_TIMES=

#model info
MODEL_ID=
REFERENCES=
SOURCE=
CALENDAR=
GRID_FILE=
MOD_TAB_DIR=
MOD_TAB=

#user info
INSTITUTE_ID=
INSTITUTION=
CONTACT=
TABLE_DIR=
```

GA-cmor3.json

```
CORDEX_EUR-11_grid.nc

rlat=412
rlon=424
float lon(rlat, rlon) ;
float lat(rlat, rlon) ;
float rlon(rlon) ;
float rlat(rlat) ;
```

2) Data ingest and data request

jupyter notebooks

From „mouth to mouth ventilation“ towards a consistent workflow with provenance

DKRZ Interactive Data Forms — Create_Submission_Form Last Checkpoint: 11/09/2016 (read only)

File Edit View Insert Cell Kernel Widgets Help

Markdown CellToolbar

Data Submission Forms for DKRZ data ingest requests

To generate a data submission form for you, please edit the cell below to include your name, email as well a your data belogs to

Then please press *"Shift" + Enter* to evaluate the cell

a link to the newly generated data submission form will be provided

please follow this link to edit your personal form

Currently the following ingest requests are supported:

- **"CORDEX"**: CORDEX data ingest requests - CORDEX data to be published in ESGF, the form is aligne original CORDEX data ingest exel sheet used for ingest requests at DKRZ
- **"CMIP6"**: CMIP6 data ingest request form for data providers - CMIP6 data to be ingested and publishe and which will be long term archived as part of the WDCC
- **"ESGF_replication"** : CMIP6 data request form for data users - request for CMIP6 data to be replicate available as part of the DKRZ national archive
- **"DKRZ_CDP"**: data ingest request for (CMIP6 related) data collections to be included in the DKRZ CMI (CDP) e.g. for model evaluation purposes
- **"test"**: for demo and testing puposes

In [ ]:
```
# please edit the (red) information below: Name, email and project the data belongs to
from dkrz_forms import form_handler

my_first_name = "...."   # example: my_first_name = "Alf"
my_last_name =  "...."   # example: my_last_name = "Mitty"

my_email =      "...."   # example: email = "alf.mitty@gmail.com"

my_project =    "...."   # available alternatives: CORDEX","CMIP6","DKRZ_CDP", "test"

my_keyword = "...."      # please select and remember your personal keyword to identify

sf = form_handler.generate_submission_form(my_first_name,my_last_name,my_email,my_projec
```

Example synda calls

Explore Metadata

example synda calls to search and explore metadata

Logout

DKRZ Interactive Data Forms — ESGF_replication_kindermann_ESMVal-Req1 Last Checkpoint: 11/04/2016 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help — Python [conda env:forms2]

Markdown CellToolbar

CMIP5 replica request form

This form is intended to request data to be replicated from other CMIP6 ESGF nodes to be made locally available in the DKRZ CMIP data pool.

The specification of a requested data collection is based on the search facets describing the data collection. These facets correspond directly to the search categories you use to find data collection in one of the ESGF portals (e.g. https://esgf-data.dkrz.de/).

The specification is based on "selection files", which are then used by the synda ESGF download application. For details please see https://github.com/Prodiguer/synda/blob/master/sdt/doc/selection_file.md

- a set of example selection files is accessible at: https://github.com/Prodiguer/synda/tree/master/sdt/selection/sample

Example selection file specification

generates a selection file named "test1.txt"

```
%%writefile test1.txt
project="CMIP5"
model="CNRM-CM5 CSIRO-Mk3-6-0"
experiment="historical amip"
ensemble="r1i1p1"
variable[atmos][mon]="tasmin tas psl"
variable[ocean][fx]="areacello sftof"
variable[land][mon]="mrsos,nppRoot,nep"
variable[seaIce][mon]="sic evap"
variable[ocnBgchem][mon]="dissic fbddtalk"
```

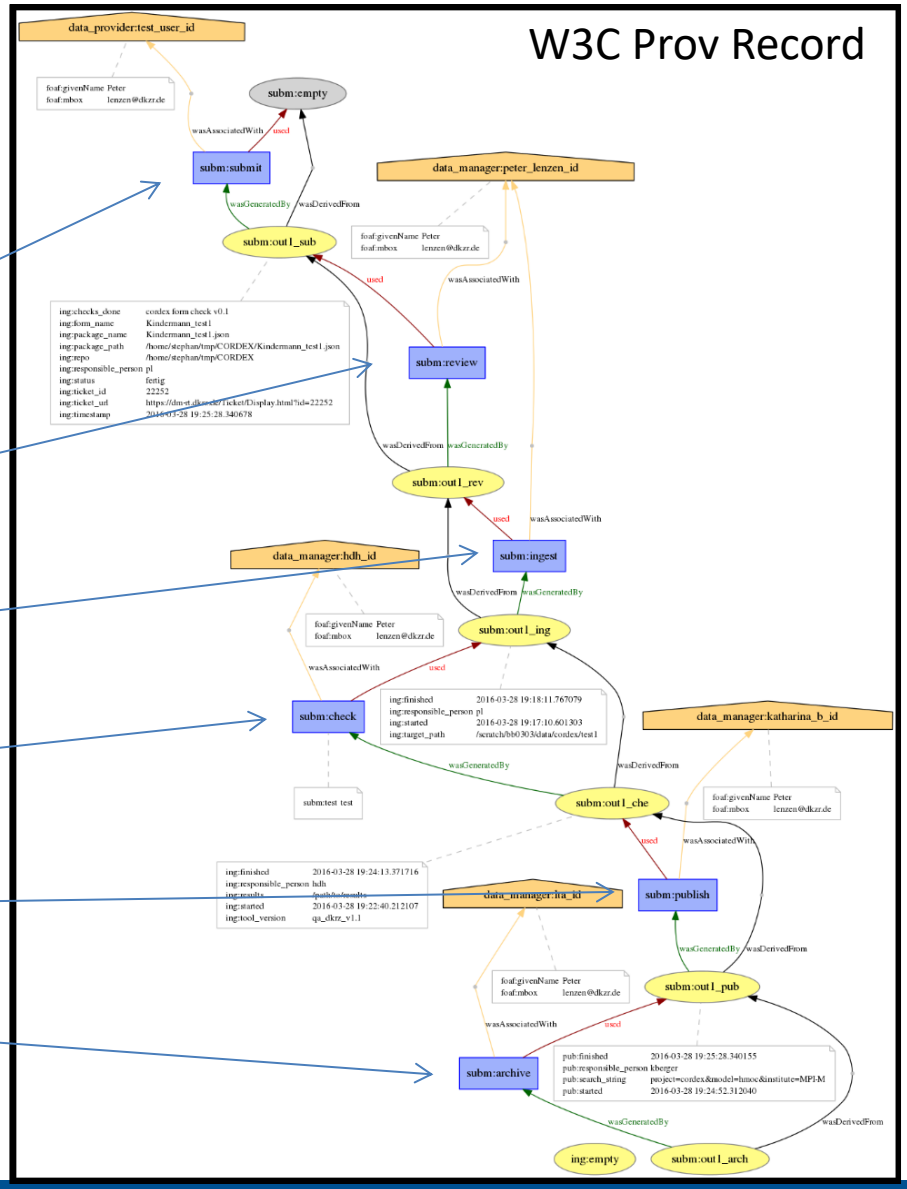
In [ ]:
```
from dkrz_forms import form_handler
from dkrz_forms import checks
my_email = "kindermann@dkrz.de"       # example: sf.email = "Mr.Mitty@yahoo.com"
my_first_name = "stephan"               # example: sf.first_name = "Harold"
my_last_name = "kindermann"             # example: sf.last_name = "Mitty"
my_keyword = "ESMVal-Req1"            # example: sf.keyword = "requestname-version"

sf = form_handler.init_form("ESGF_replication",my_first_name,my_last_name,my_email,my_keyword)
```

Recording the data ingest related workflow

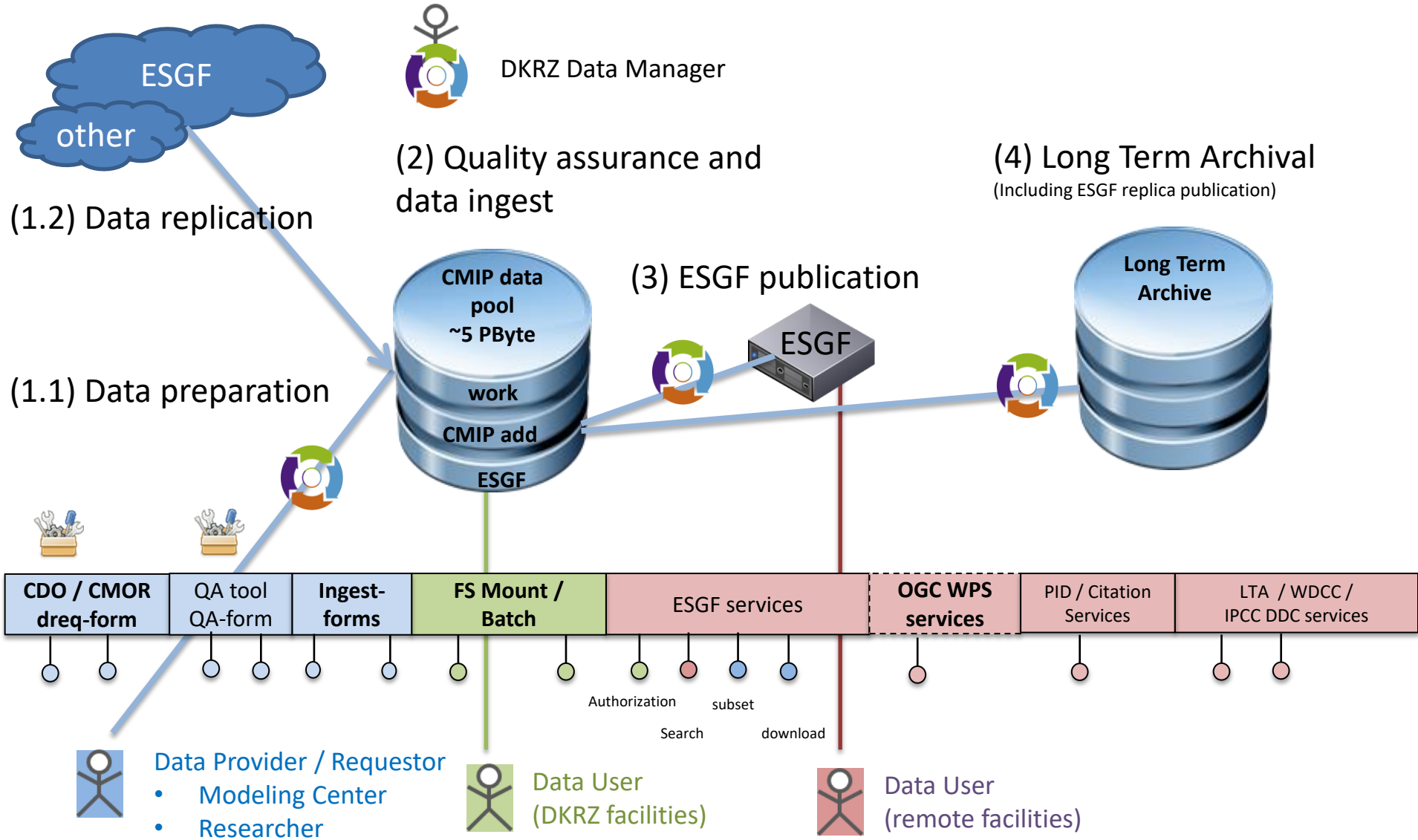
All data ingest related activities are recorded in json files

Various tools to handle and evaluate these json files

- Data submission form
- Request review
- Data ingest
- Data quality assurance
- ESGF publicaton
- Archival / ESGF tape publication



ESGF / CMIP data management workflow and services



WPS processing

- DKRZ initiated „birdhouse“ open source WPS ecosystem
  - On github, CI tests and docu
  - Highly modular design (GUI, WPS backend, codes)
  - Modular installation and deployment (conda packaging, docker images)
- Birdhouse will be used at DKRZ to provide WPS services (ESGF-CWT defined, dedicated)
- Birdhouse will be used at DKRZ, IPSL and BADC to provide Copernicus WPS services (CMIP5/6, Cordex)
  - Key challenges: code packaging and deployment procedure for WPS codes as well as scalability

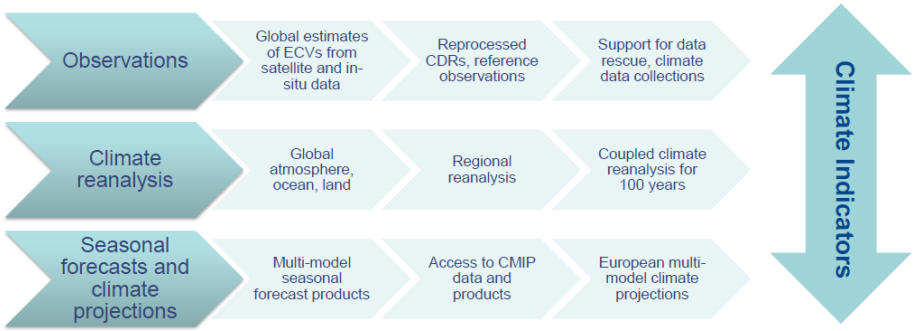
CMIP WPS Processing and Copernicus

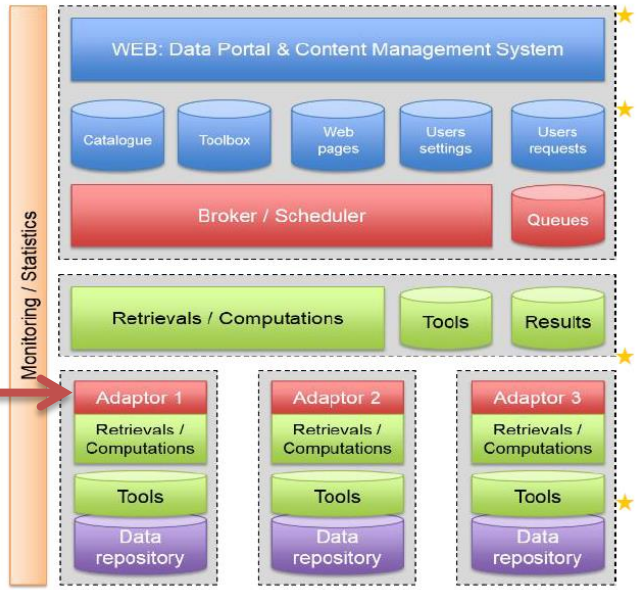
- European Copernicus Climate Change Service (C3S) implemented by ECMWF (operational from 2018 on)

  - Authoritative source of climeat information for EuropeClimate
  - Builds upon national investments and national climate service providers

  - Climate Data Store (CDS) will be heart of C3S infrastructure

    - WPS interfaces for code deployed near European ESGF sites
    - Birdhouse ecosystem deployments at DKRZ,IPSL, BADC

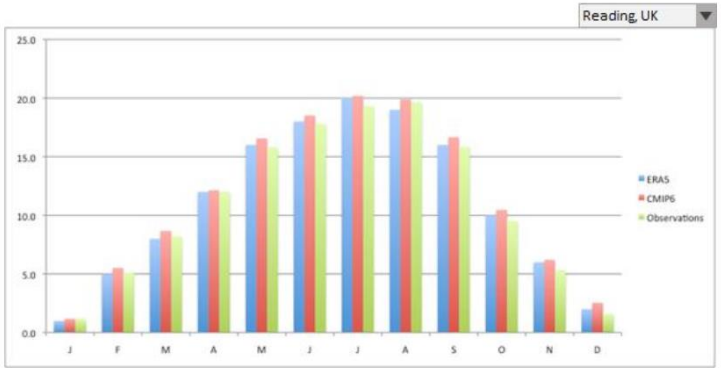


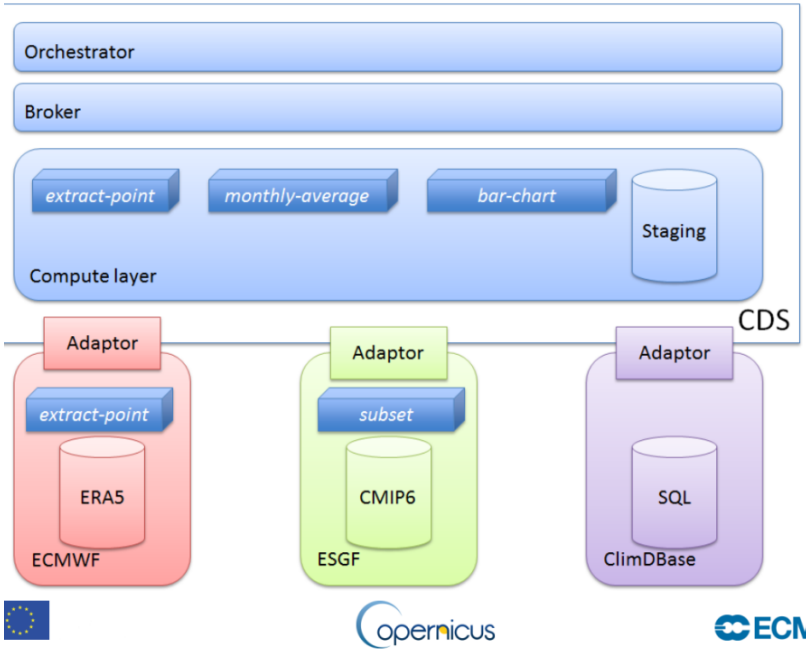


C3S example

Hypothetical example: monthly average of temperature at selected location

- ERA5: 40 years reanalysis, hosted at ECMWF (GRIB, in Kelvin)
- CIMP6: 2000 years climate projections, hosted in an ESGF node (NetCDF, Kelvin)
- Observation: time series of temperature measured at a given station, hosted in ClimatDBase (SQL, imaginary dataset)





Copied from: [Boudouin Raoults Presentation](https://www.rd-alliance.org/sites/default/files/attachment/Copernicus_C3S_CDS_BaudouinRaoult.pdf)

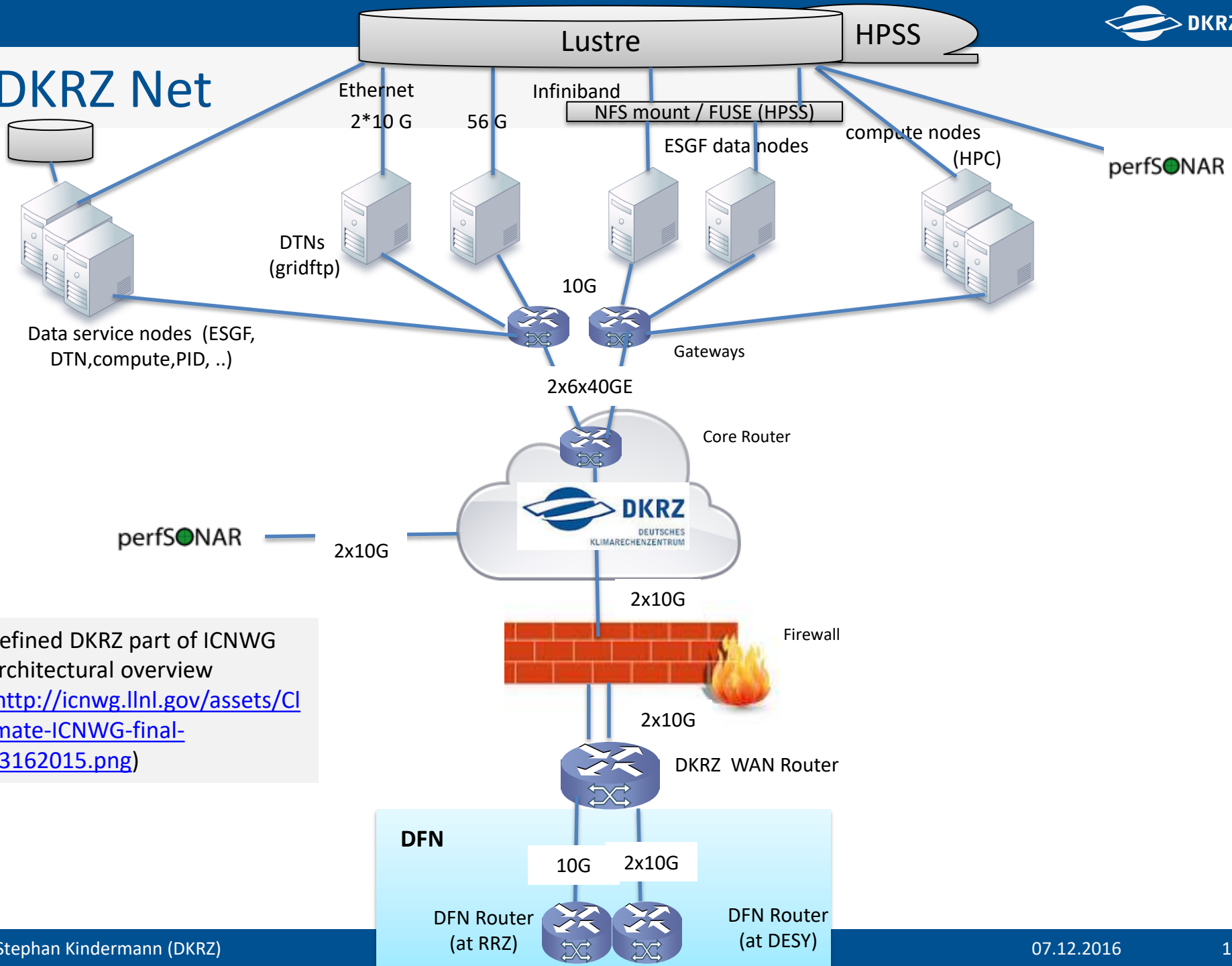
Summary

- DKRZ CMIP data pool (with tape backend)
  - cdo/cmor support, replica, ESGF, WPS
  - Processing aspect very important in future
    (→ data center responsibility: provide provenance info ..)

- PID service (→ Merret‘s talk)
- IPCC DDC, Data citation service (→ Martina‘s talk)
- Quality assurance tool / service (→ Heinz Dieter‘s talk)
- Birdhouse WPS ecosystem (→ Stephan‘s talk)

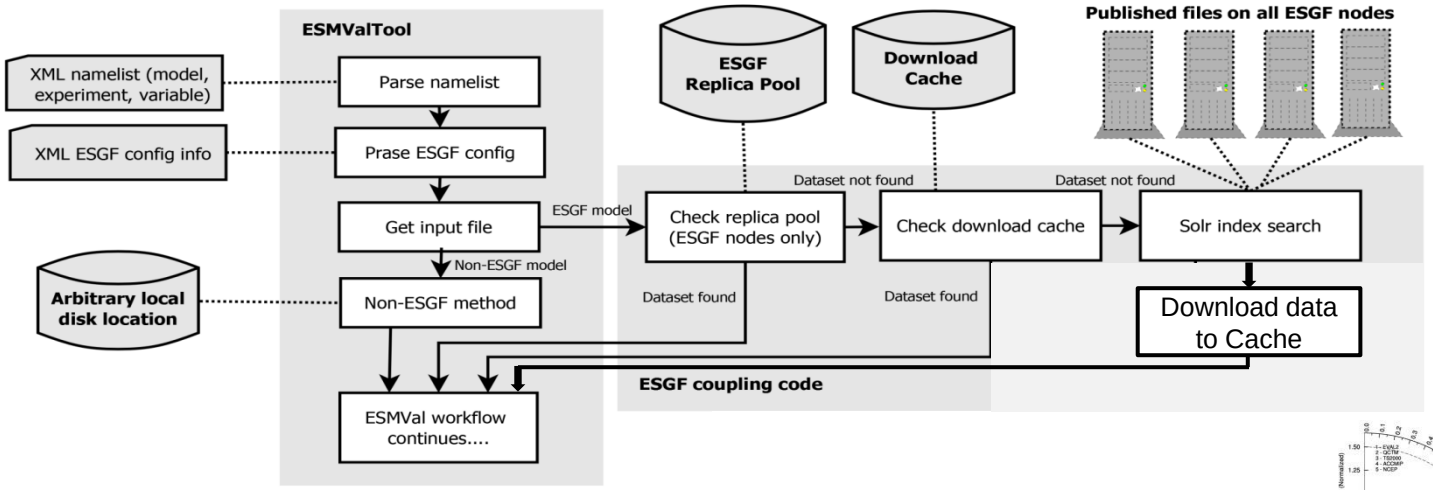
Additional slides

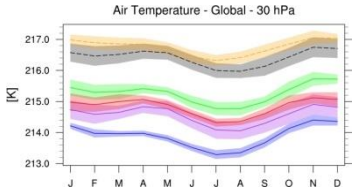
DKRZ Net

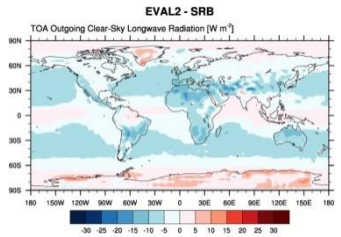


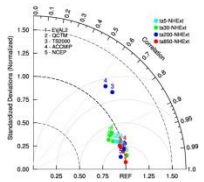
Refined DKRZ part of ICNWG architectural overview (http://icnwg.llnl.gov/assets/Climate-ICNWG-final-03162015.png)

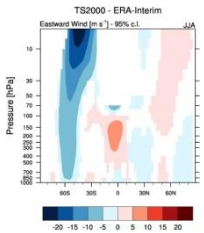
CMIP6 evaluation

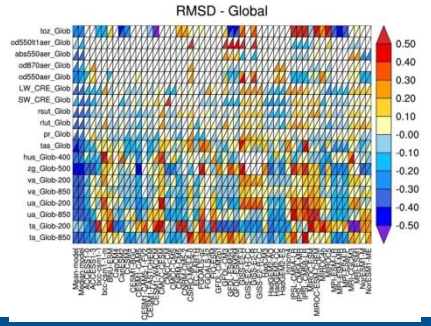










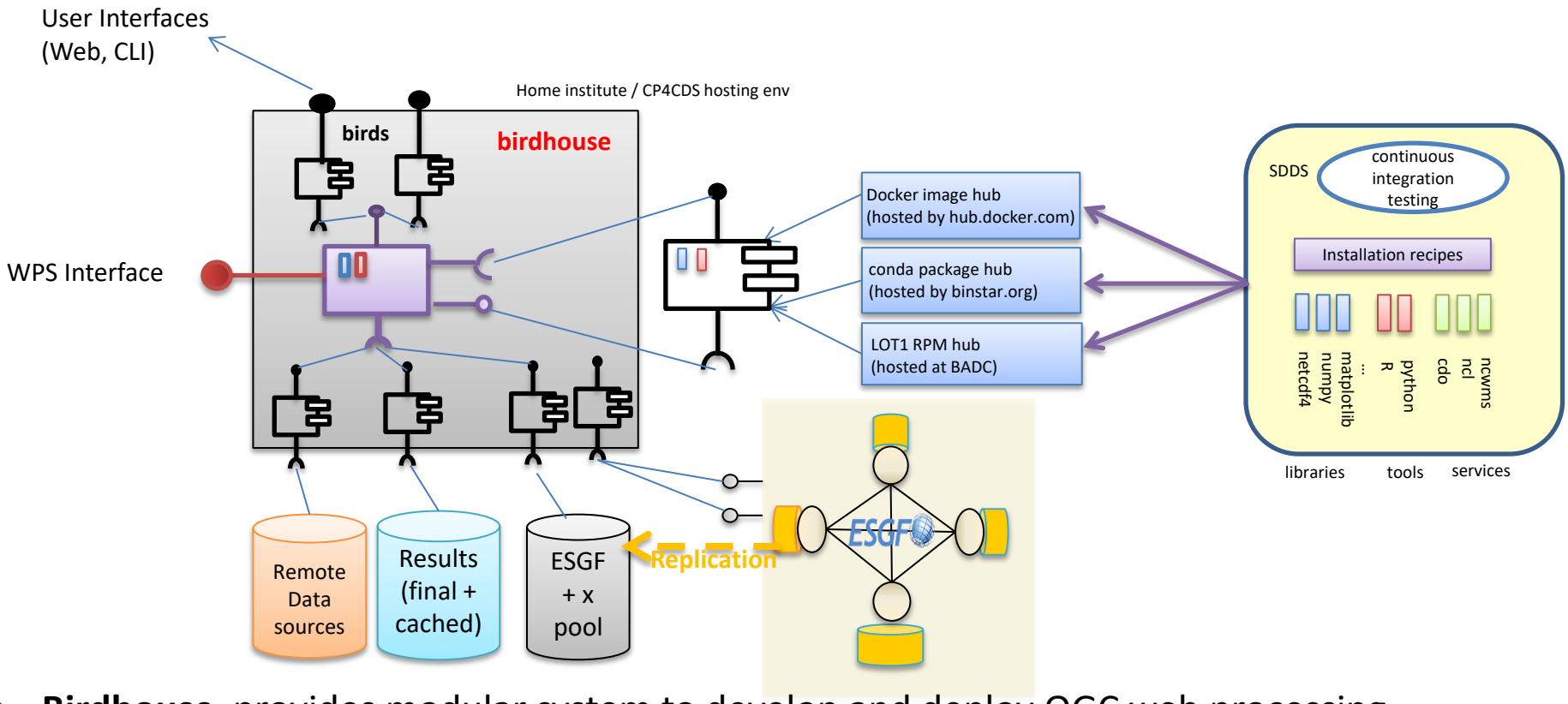


www.esmvaltool.org

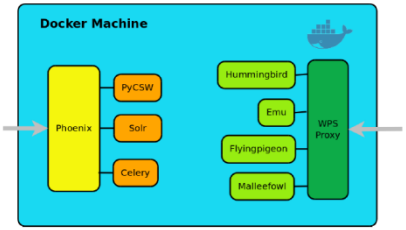
Eyring et al., Geosci. Model Dev., 2016

www.github.com/ESMValTool-Core/ESMValTool

Compute node software: Overview



- **Birdhouse** provides modular system to develop and deploy OGC web processing services
  - code, recipes: https://github.com/bird-house
  - binstar channel: https://conda.anaconda.org/birdhouse,
  - Docker hub: https://hub.docker.com/u/birdhouse
  - documentation: http://birdhouse.readthedocs.org
  - Demo installation: http://mouflon.dkrz.de



The Birdhouse approach