Status of WP4: Exploitability

Julian Kunkel[1,(7)] Bryan N. Lawrence[2,3] Jakob Luettgau[7] Neil Massey[4]
Alessandro Danca[5] Sandro Fiore[5]
Huang Hu[6]

1 Department of Computer Science, University of Reading
2 UK National Centre for Atmospheric Science
3 Department of Meteorology, University of Reading
4 STFC Rutherford Appleton Laboratory
5 CMCC Foundation
6 Seagate Technology LLC
7 DKRZ

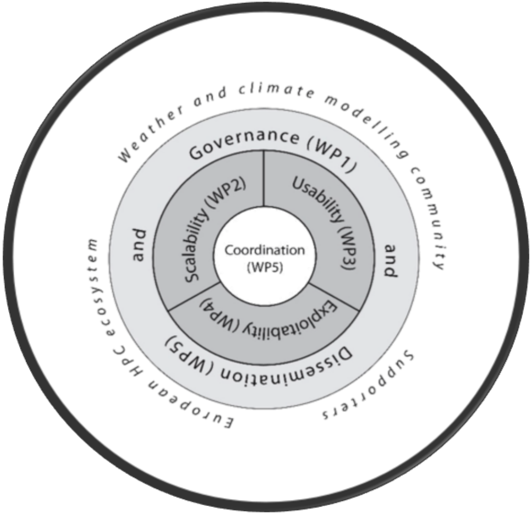
6 November 2018

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

Outline

Disclaimer: This material reflects only the author's view and the EU-Commission is not responsible for any use that may be made of the information it contains

Project Organisation



WP1 Governance and Engagement

WP2 Global high-resolution model demonstrators

WP3 Usability

WP4 Exploitability

- The business of storing and exploiting high volume data
- Storage layout for Earth system data
- Methods of exploiting tape

WP5 Management and Disssemination

Work Package 4 — Exploitability (of data)

Partners

DKRZ, STFC, CMCC, Seagate, UREAD

ECMWF was a partner but we removed the relevant task in the reprofiling following the first review

Task 4.1

Business models

- Documentation
  Coarse-grained model
  Fine-grained model
- D4.1 completed
- **Task is completed**

Task 4.2

New Storage Layout

- Software & Design
  ESD Middleware
- Design delivered D4.2
- Initial benchmarks
- Development ongoing

Task 4.3

New Tape Methods

- Software
  JDMA data migration
- Prototype in place
- Wrapup ongoing

Outline

Coarse-Grained Models

Simple graph models

High-level representation

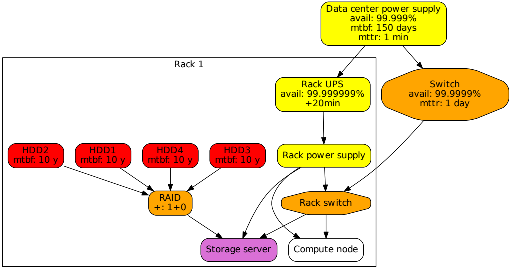
- Hardware/software
- Purpose: Ease understanding

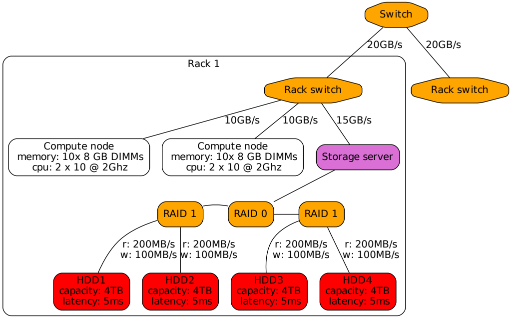
Includes:

- performance
- resilience
- cost

Deliverable D4.1 (done)

Scenarios discussing architectural changes for data centres, and implications for cost/performance

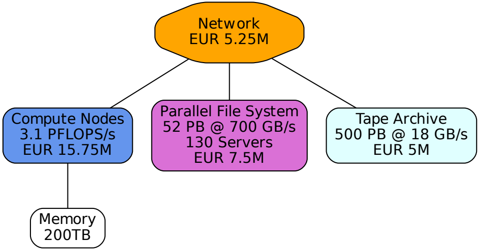




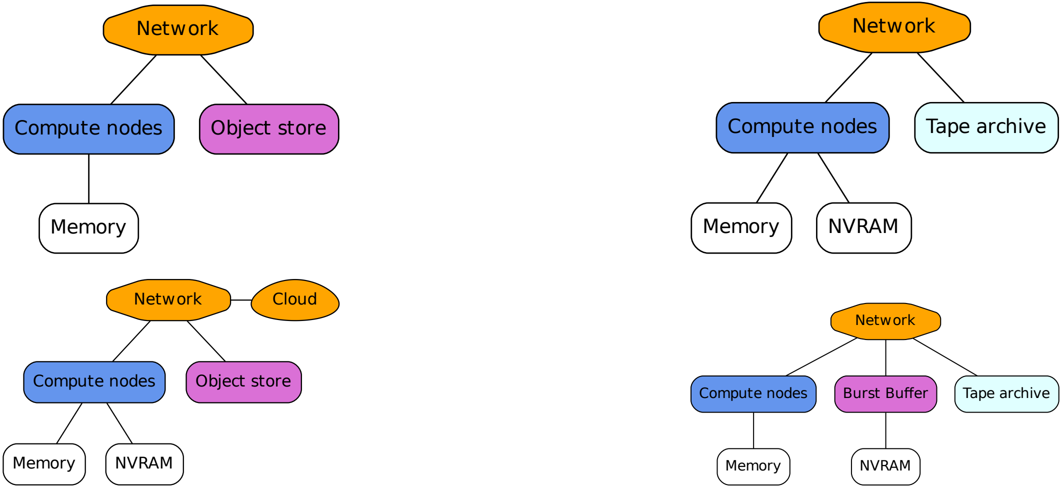
Some Examples of Business Considerations

One cost model of storage based on DKRZ

- Tape: 12 € per TB/ year
- Software licenses for tape are driving the costs!
- Parallel Disk: 28 € TB/year
- Object storage: 12.5 € TB/year (without software license costs)
- Cloud: $ 48 TB/year (only storage, access adds costs)
- Alternative models: 8 € / 153 € for tape/disk per year
- Idle (unused) data is an important cost driver!



Alternative Storage Landscapes



Fine-Grained Performance Modelling

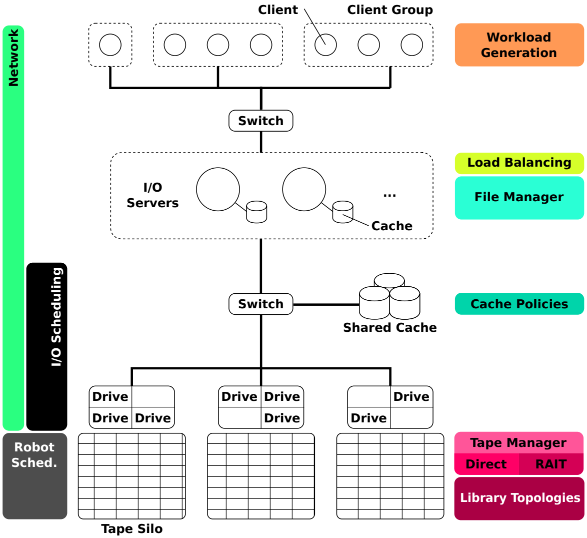
Detailed Modelling

A simulator has been developed; covers

- HW, software, tape drives, library, cache
- Can replay recorded FTP traces
- Validated with DKRZ environment

Usage

Aim to use to evaluate performance and costs of future storage scenarios – particularly tape



Outline

Dealing with Climate/Weather Data

Challenges in the domain of climate/weather

- Large data volume and high velocity
- Data management practice does not scale & not portable
  - Difficult to manage file placement / knowledge of content
  - Hierarchical namespaces do not reflect use cases
  - Individual solutions at every site
- Suboptimal performance & performance portability
  - Cannot properly exploit the hardware / storage landscape
  - Tuning file formats and file sytem necessary at *application* level
- Data conversion is often needed
  - To combine data from multiple experiments, time steps, ...

Earth-System Data Middleware

Design Goals of the Earth-System Data Middleware

1. Relaxed access semantics, tailored to scientific data generation
   - Avoid false sharing (of data blocks) in the write-path
   - Understand application data structures and scientific metadata
   - Reduce penalties of **shared** file access
2. Site-specific (optimized) data layout schemes
   - Based on site-configuration and performance model
   - Site-admin/project group defines mapping
   - Flexible mapping of data to multiple storage backends
   - Exploiting backends in the storage landscape
3. Ease of use and deployment particularly configuration
4. Enable a configurable namespace based on scientific metadata

Benefits

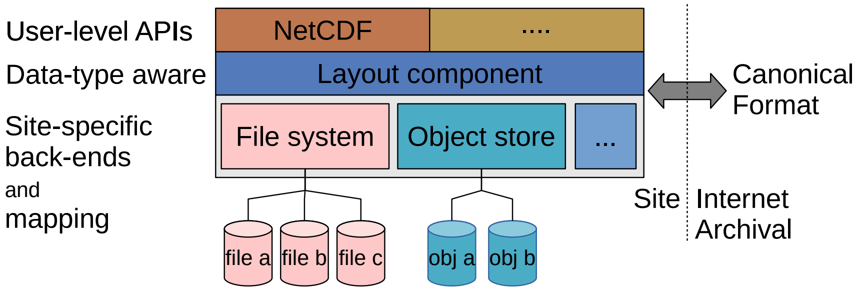
- Independent, share-nothing lock-free writes from parallel applications
- Storage layout is optimized to local storage
  - Exploits characteristics of diverse storage
  - Preserve compatibility by creating platform-independent file formats on the site boundary/archive
- Less performance tuning from users needed
  - One data structure can be fully or partially replicated with different layouts
  - Using multiple storage systems concurrently
- (Expose/access the same data via different APIs[1])
- (Flexible and automatic namespace[1])

[1] Not shown in ESiWACE scope

Architecture

Key Concepts

- Middleware utilizes layout component to make placement decisions
- Applications work through existing API (currently: NetCDF library)
- Data is then written/read efficiently; potential for optimization inside library



ESDM Status

Status

- ESDM Architecture Design for Prototype (Deliverable D4.2)
- Multi-threaded data path
- Data backend Plugins for POSIX, CLOVIS, WOS
  - Reached: MS7 Prototypes of alternative storage backends
- Trivial metadata store on the shared file system
- 50%: HDF5 VOL plugin as application to ESDM adapter
  - Proof of concept for adaptive tier selection in HDF5
- 40%: ESDM core implementation as library
- Evaluation of **ESDM benchmark** at DKRZ, STFC, CMCC
  - Reached: MS9 Implementation of ESD middleware at STFC and CMCC

Evaluation of the Prototype: Here at DKRZ Mistral

System

- Test system: DKRZ Mistral supercomputer
- Nodes: 200

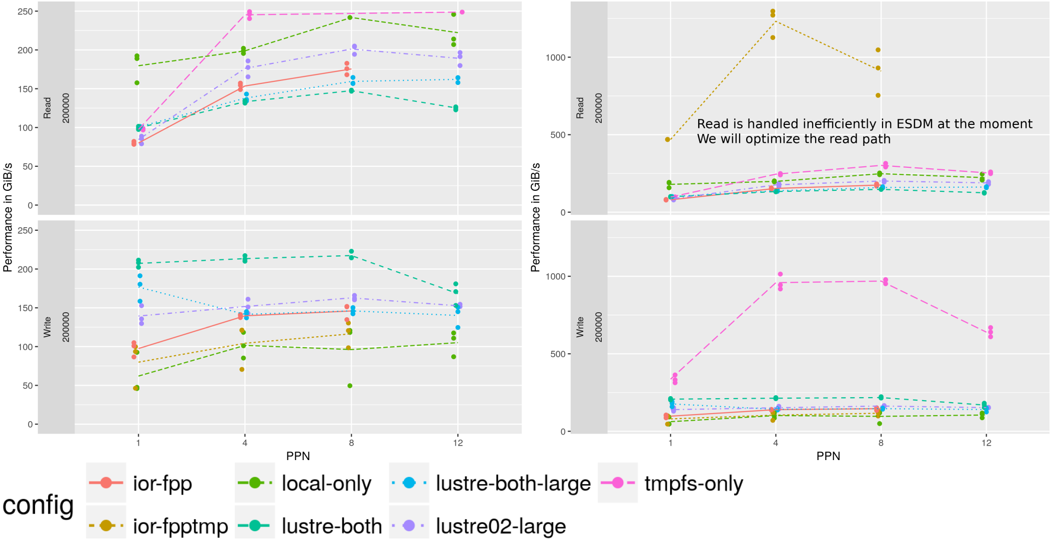
Benchmark

- Uses ESDM interface directly; Metadata on Lustre
- Write/read a timeseries of a 2D variable
- Grid size: $200k \cdot 200k \cdot 8$ Byte $\cdot$ 10 iterations
- Data volume: size $=$ 2980 GiB; compared to IOR performance

ESDM Configurations

- Splitting data into fragments of 100 MiB (or 500)
- Use different storage systems

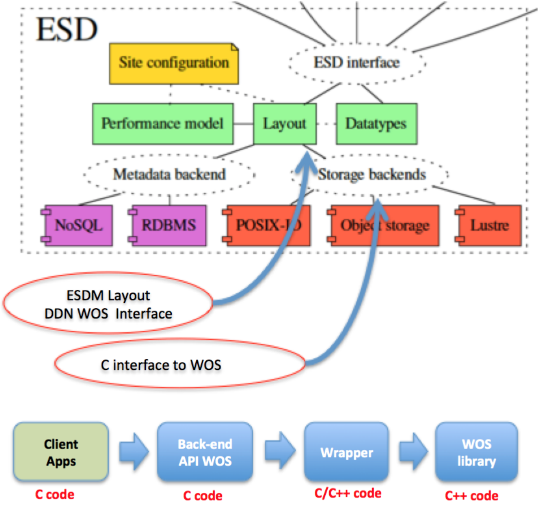
Measured Performance



Data Backends – DDN Object Store (CMCC)

WOS Prototype

- Backend works
- Developed C wrapper for the C++ DDN WOS libraries
- Designed a parallel approach for independent / multiple write operations on WOS storage
- Problem: WOS is discontinued!

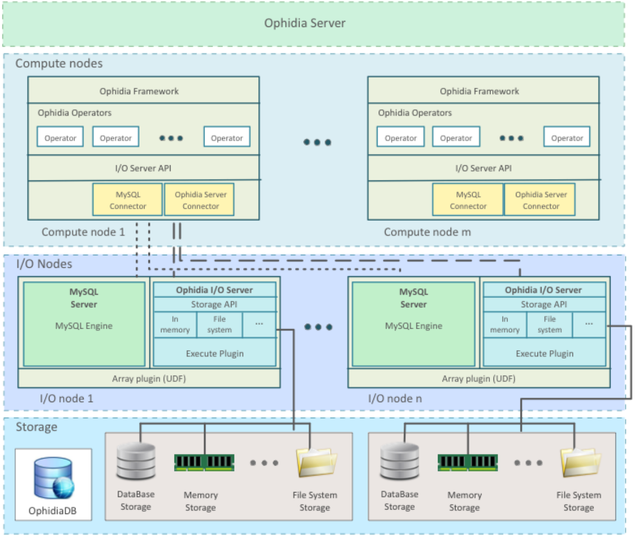


Deployment Testing Example

Test and Deployment

Ophidia (in-memory data analytics) as a test application for ESDM

- **Import** and **Export**
  Ophidia operators adapted for integration with ESDM storage
  - Uses patched NetCDF
- ESDM successfully built on:
  - Athena HPC Cluster
  - OphidiaLab
- Creation of a VM for the whole software stack



Task Roadmap

Roadmap until the end of ESiWACE1

- Supporting a subset of NetCDF applications
  - NetCDF benchmark
  - Ophidia: use in a big data workflow
  - Toy model: Shallow water equation
- Improve data plugins
- Improve data layouting
- Optimize read path
- Run benchmarks at sites
  - CLOVIS performance in various configurations on a reasonable cluster
- Build a performance model for WOS (and CLOVIS) as blueprint for other backends

Outline

Approach

Semantic Storage Library

Task 3: Developing new tape access strategies and software . . . higher bandwidth to tape storage and increased storage redundancy.

- ~~Increase bandwidth to/from tape by exploiting RAID-to-TAPE~~.
  - Decided that this was too difficult to do in a portable manner and that portable (tape + object store) workflow was a more important initial priority.
- Provide a portable library to address user management of data files on disk (POSIX and/or Object Store) and tape which
  1. does not *require* significant sysadmin interaction, but
  2. can make use of local customisation if available/possible
  3. exploits existing metadata conventions
  4. can eventually be backported to work with the ESDM
  5. prototype can be deployed fast enough that we can use it for Exascale Demonstrator

Architecture

Two Key Components

1. S3NetCDF — replacement for NetCDF4-python with support for object stores
2. CacheFace — a portable frontend for managing content in object stores/tape

Architecture

Two Key Components

1. S3NetCDF — replacement for NetCDF4-python with support for object stores
2. CacheFace — a portable frontend for managing content in object stores/tape

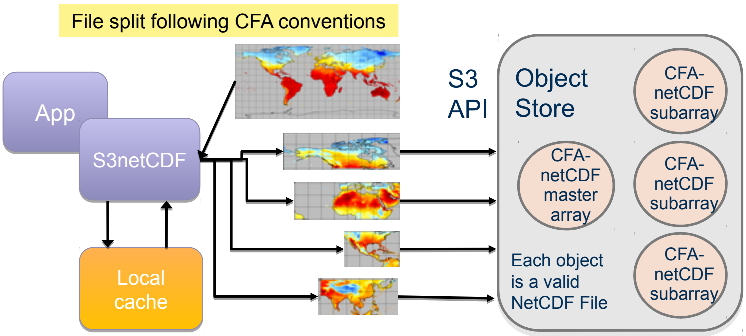
Information Structure

Exploiting the Climate Forecast Aggregation (CFA) Framework[1], which

1. Defines how multiple CF fields may be combined into one larger field (or how one large field can be divided)
2. Is fully general and based purely on CF metadata
3. Includes a syntax for storing an aggregation in a NetCDF file using **JSON** string content to point at aggregated files

[1]:https://goo.gl/DdxGtw

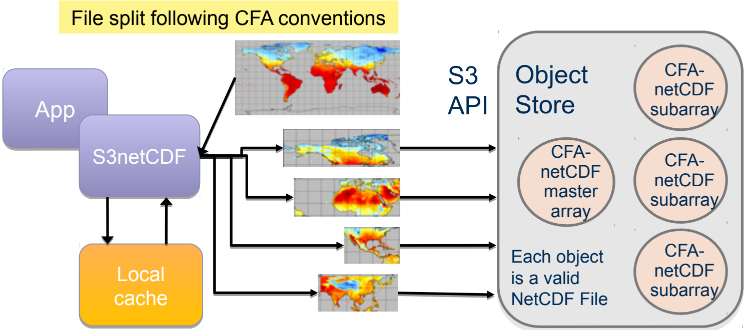
S3NetCDF (working title)



Architecture

- Master Array File is a NetCDF file containing dimensions and metadata for the variables including URLs to fragment file locations
- Master Array file optionally in persistent memory or online, nearline, etc
  NetCDF tools can query file CF metadata content without fetching them

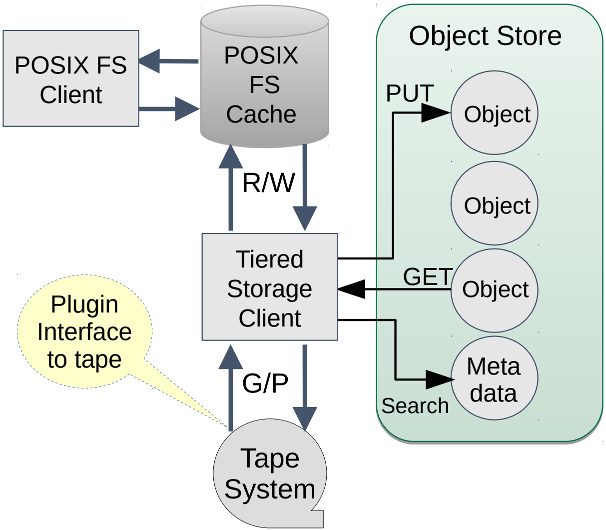
S3NetCDF (working title)



Status:

- Prototype released (milestone 7B). Subsequent refactoring complete (October 2018) in preparation for parallelisation.
- ESiWACE1 goal: add prototype parallelisation, measure performance, publish paper and more complete usage documentation. (ESiWACE2: performance, integrate components with ESDM).

CacheFace (working title)

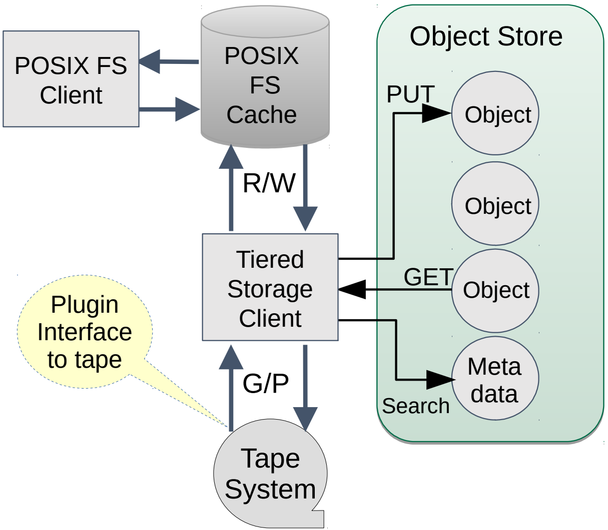


Architecture

Three key components:

1. a cache management utility,
2. a data migration utility,
3. and a metadata system.

CacheFace (working title)



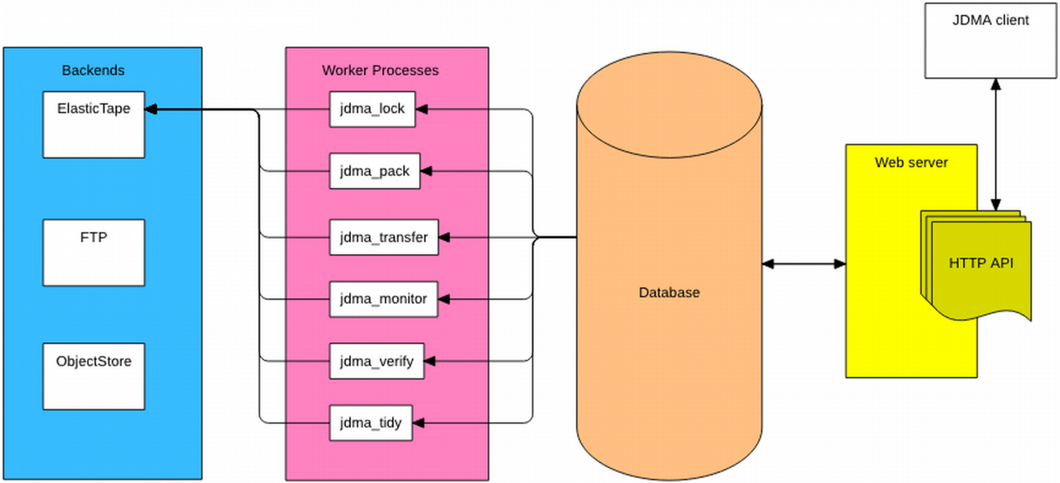
Status

- Simple metadata system designed.
- Cache system designed and prototype built that can use Minio interface to object store.
- Data migration prototype (JDMA, next slides) developed with support for tape (milestone 8) and object store (soon) and about to be deployed operationally for Elastic Tape backend (on JASMIN).
- EsiWACE1 goals: complete JDMA, extend and test backends, (ESiWACE2: Finalise metadata and cache systems, integrate components with ESDM).

JDMA: a Prototype Tape Library for Advanced Tape Subsystems

- JDMA: JASMIN Data Migration App(lication)
- A multi-tiered storage library
  - Provides a single API to users to move data to and from different systems
  - HTTP API running on webserver, database records requests and file metadata
  - Command line client which interfaces to HTTP API
- Multiple storage "backends" supported via plugin
  - Amazon S3 (Simple Storage Solution) for Object Stores and AWS
  - FTP, also for tape systems with a FTP interface
  - Elastic Tape – a proprietary tape system based on CASTOR
- A number of daemons (scheduled processes) carry out the data transfer
  - Asynchronously
  - On behalf of the user

JDMA System Architecture



Outline

Summary

Current Status

1. Business: Complete
2. ESDM: Architecture and prototypes exist with multiple backends.
3. SemSL: Architecture and prototypes exist
   - S3NetCDF initially targeting object stores
   - CacheFace, initially targeting tape

ESiWACE1 Goals

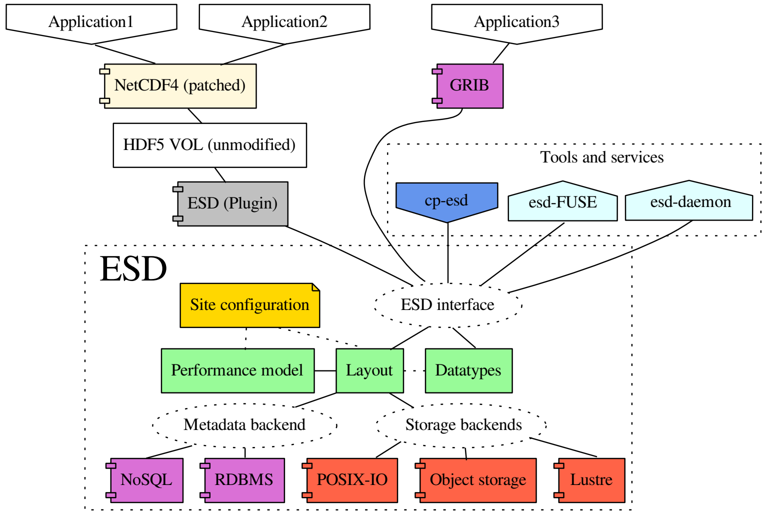
1. ESDM: Extend use exemplars, improve plugin, layout, and performance components for multiple backends
2. SemSL: S3NetCDF – parallelise and publicises; CacheFace – Deploy JDMA. Release prototype complete system.

The ESiWACE project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No **675191**

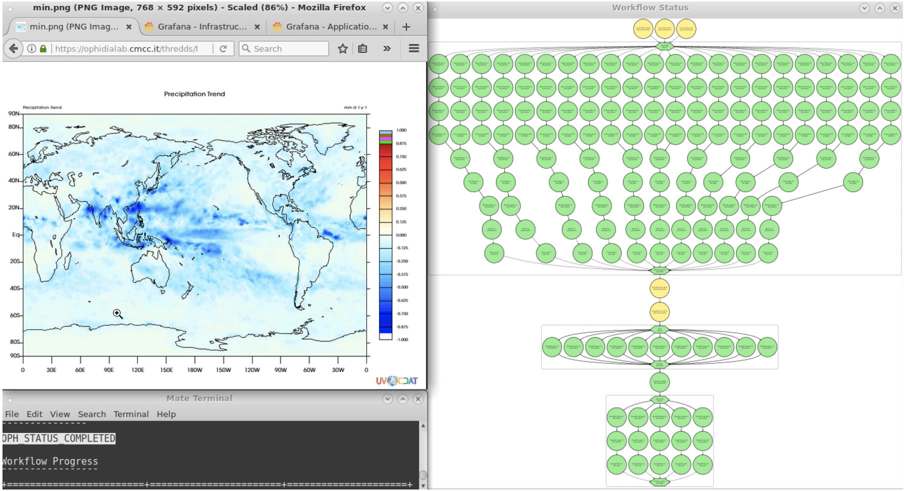
esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

Disclaimer: This material reflects only the author's view and the EU-Commission is not responsible for any use that may be made of the information it contains

Architecture: Detailed View of the Software Landscape



Ophidia Example BigData Workflow (See WP3 D3.10)



The PTA multi-model workflow implemented in Ophidia has been executed and validated at CMCC on 11 models from CMIP5 experiment for a total of 181 tasks, 2.5 minutes, 96 cores on OphidiaLab