Computer Science Department

HPS

University of Reading

The Need for Next Generation Semantic Interfaces to Process Climate/Weather Workflows

Limitless Storage
Limitless Possibilities

https://hps.vi4io.org

Julian M. Kunkel

SIG IO UK

2018-06-06

Outline

The Software Stack for NWP/Climate

University of Reading

- Domain semantics
  - XIOS writes independent variables to one file each
  - 2nd servers for performance reasons
  - Why do we need parallel file systems here?
  - Why can't the middleware do appropriate data shuffling?
- Data model in the middleware NetCDF4/HDF5
  - Performant mappings to files are limited
    - Map data semantics to one "file"
    - File format notorious inefficient
  - Domain metadata is treated like normal data
    - Need for higher-level databases like Mars
  - Interfaces focus on variables but lack features
    - Workflows
    - Information life cycle management

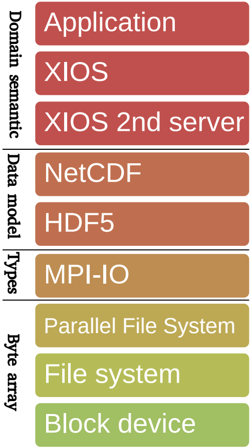


Figure: Typical I/O stack

Critical Discussion

University of Reading

Questions from the users' perspective

- Why do I have to organize the file format?
  - It's like taking care of the memory layout of C-structs
- Why do I have to convert data between storage paradigms?
- Why must I provide system specific performance hints?
  - It's like telling the compiler to unroll a loop exactly 4 times
- Why is a file system not offering the consistency model I need?
  - My application knows the required level of synchronization
- Why can't I rely on a correct implementation of the consistency model?
  - Parallel file systems have performance issues with most models

Being a user, I would rather code an application?

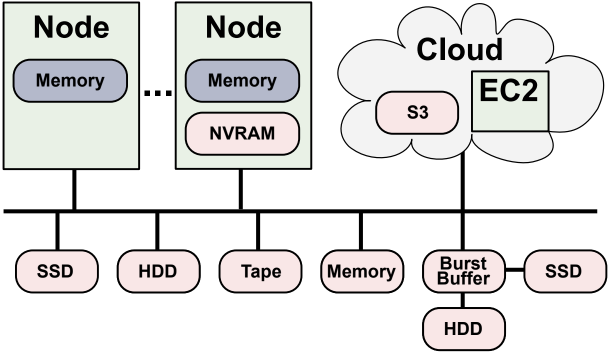
Challenges Faced by HPC I/O

University of Reading

- Difficulty to analyze behavior and understand performance
  - Unclear access patterns (users, sites)
- Coexistence of access paradigms in workflows
  - File (POSIX, ADIOS, HDF5), SQL, NoSQL
- Semantical information is lost through layers
  - Suboptimal performance, lost opportunities
  - All data treated identically
- Reimplementation of features across stack
  - Unpredictable interactions
  - Wasted ressources
- Restricted (performance) portability
  - Optimizing each layer for each system?
  - Users lack technological knowledge for tweaking
- Utilizing the future storage landscapes
  - No performance awareness, manual tuning and tiering needed

Future Systems: Coexistence of Storage Systems

University of Reading



- We shall be able to use all storage technologies concurrently
  - Without explicit migration etc. put data where it fits
  - Administrators just add a new technology (e.g., SSD pool) and users benefit
  - Should be steered by a standard and open interface
  - Open ecosystem for any vendor...

Outline

Compression Research: Involvement

University of Reading

- **Scientific Compression Library** (SCIL)
  - Separates concern of **data accuracy** and **choice of algorithms**
  - Users specify necessary accuracy and performance parameters
  - Metacompression library makes the choice of algorithms
  - Supports also new algorithms
  - Ongoing: standardization of useful compression quantities
- Development of algorithms for lossless compression
  - MAFISC: suite of preconditioners for HDF5, pack data optimally, reduces climate data by additional 10-20%, simple filters are sufficient
- Cost-benefit analysis: e.g., for long-term storage MAFISC pays of
- Analysis of compression characteristics for earth-science related data sets
  - Lossless LZMA yields best ratio but is very slow, LZ4fast outperforms BLOSC
  - Lossy: GRIB+JPEG2000 vs. MAFSISC and proprietary software
- Method for system-wide determination of ratio/performance
  - Script suite to scan data centers...

SCIL: Supported User-Space Quantities

Quantities defining the residual (error):

absolute tolerance: compressed can become true value ± absolute tolerance

relative tolerance: percentage the compressed value can deviate from true value

relative error finest tolerance: value defining the abs tol error for rel compression for values around 0

significant digits: number of significant decimal digits

significant bits: number of significant decimals in bits

field conservation: limits the sum (mean) of field’s change

Quantities defining the performance behavior:

compression throughput

decompression throughput

in MiB or GiB, or relative to network or storage speed

Aim to standardize user-space quantities across compressors!
See `https://www.vi4io.org/std/compression`

Ongoing Activity: Earth-Science Data Middleware

University of Reading

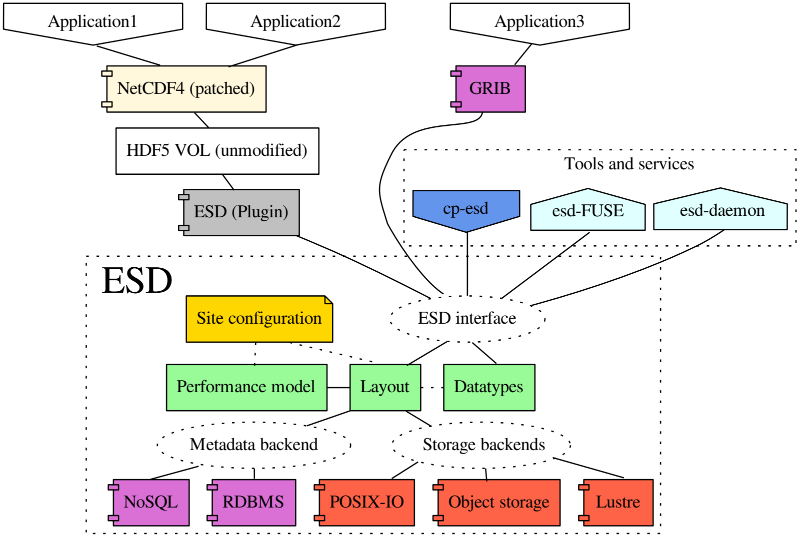
Part of the ESiWACE Center of Excellence in H2020

Design Goals of the Earth System Data Middleware

1. Understand application data structures and scientific metadata
2. Flexible mapping of data to multiple storage backends
   - Placement based on site-configuration + performance model
   - Site-specific optimized data layout schemes
3. Relaxed access semantics, tailored to scientific data generation
4. A configurable namespace based on scientific metadata

Architecture

University of Reading



Thoughts

University of Reading

I believe we must re-architect the IO stack

- Smart hardware and software components
- Storage and compute cannot be optimized individually but together
- User metadata and workflows as first-class citizens
- Self-aware instead of unconscious
- Self-learning and improving over time

Outline

A Potential Approach in the Community: Following MPI

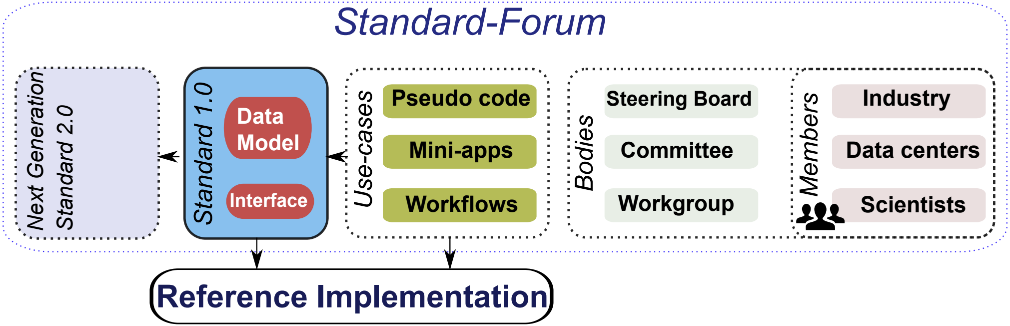
University of Reading

- The **standardization** of a high-level *data model & interface*
  - Targeting data intensive and HPC workloads
  - Lifting semantic access to a new level
- Development of a reference implementation of a **smart runtime system**
  - Implementing key features
- Demonstration of benefits on socially relevant data-intense apps

Development of the Data Model and API

University of Reading

- Establishing a Forum similarly to MPI
- Define data model for HPC
  - To have a future: must be beneficial for Big Data + Desktop, too
- Open board: encourage community collaboration

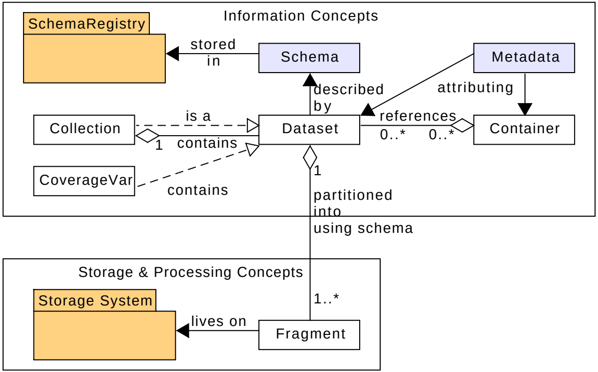


API Key Features

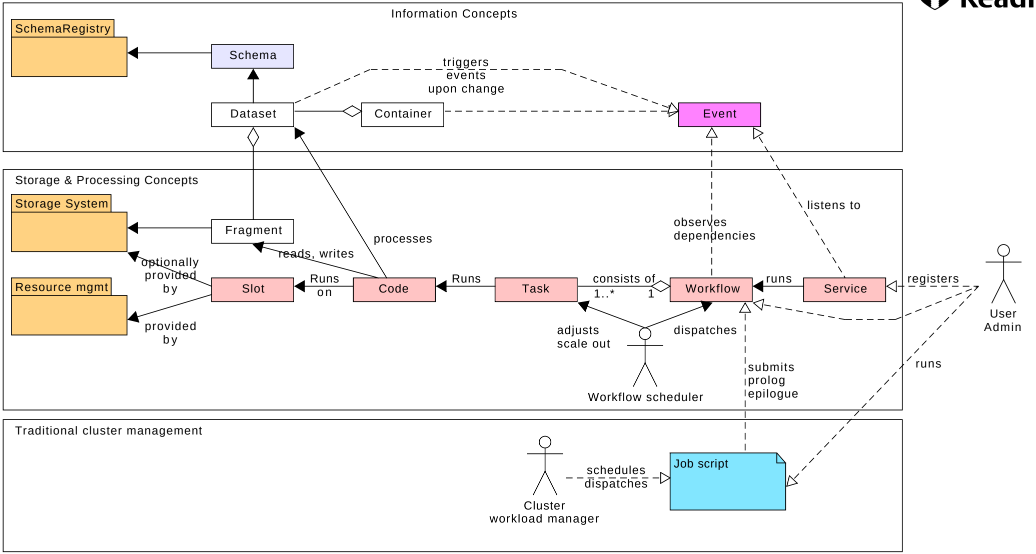
University of Reading

- High-level data model for HPC
  - Storage understands data structures vs. byte array
  - Relaxed consistency
- Semantic namespace and storage-aware data formats
  - Organize based on domain-specific metadata (instead of file system)
  - Support domain-specific opperations and addressing schemes
- Integrated processing capabilities
  - Offload data-intensive compute to storage system
  - In-situ/In-transit workflows
- Workflow management
  - Managed data-driven workflow
- Performance-portability
  - Guided interfaces: Intents vs. technical hints
- Enhanced data management features
  - Embedded performance analysis
  - Resilience, import/export, ...

Data Model: Data Formats are Provided by Storage



Processing Model



Summary

University of Reading

- The separation of concerns in the existing storage stack is suboptimal
- SCIL and ESDM are research examples towards next generation
- Semantic interfaces combined with ML will be a game changer
- Can the community work together to define next generation APIs?

Appendix

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