Brightskies

DAOS : SEIS mapping

DUG'20

Mirna Moawad*, Amr Nasr (Brightskies)

Johann Lombardi, Mohamad Chaarawi, Philippe Thierry (Intel)

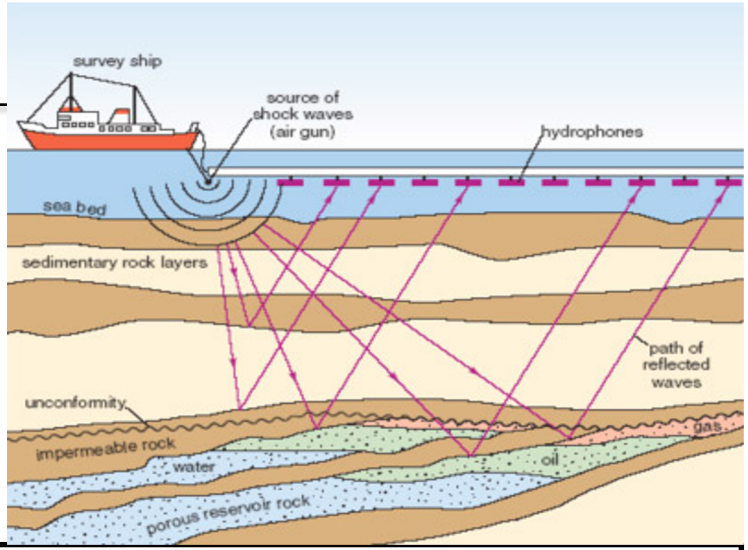
19/11/2020

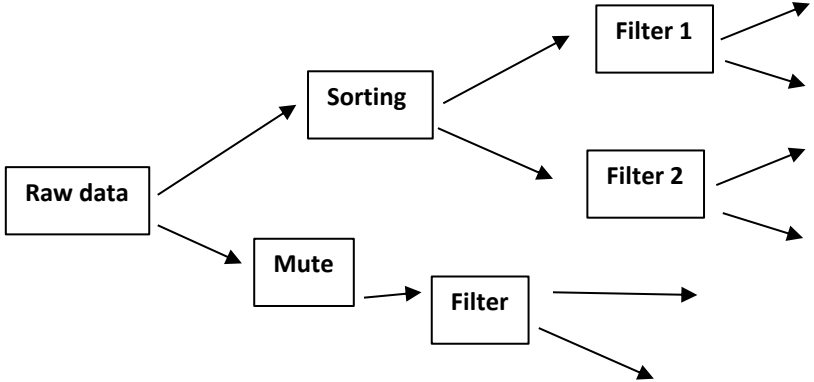
Agenda

- **Marine seismic acquisition**
- Motivation
- DAOS-SEIS mapping Graph
- DAOS-SEIS API
- Benchmarking Results

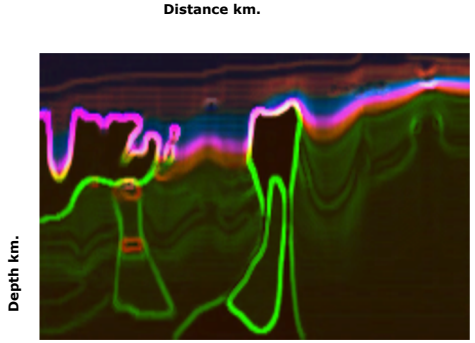
Marine seismic acquisition

- 1 shot :
  - 8 streamers*1000 receivers*5001 samples*4 bytes= 0.15 GB
- 1 line :
  - 20km / 25m = >> 800 shots per line = 120GB per line
- 2400 lines
  - **=> 280 TB of Raw seismic data** for about 1200 km² acquisition





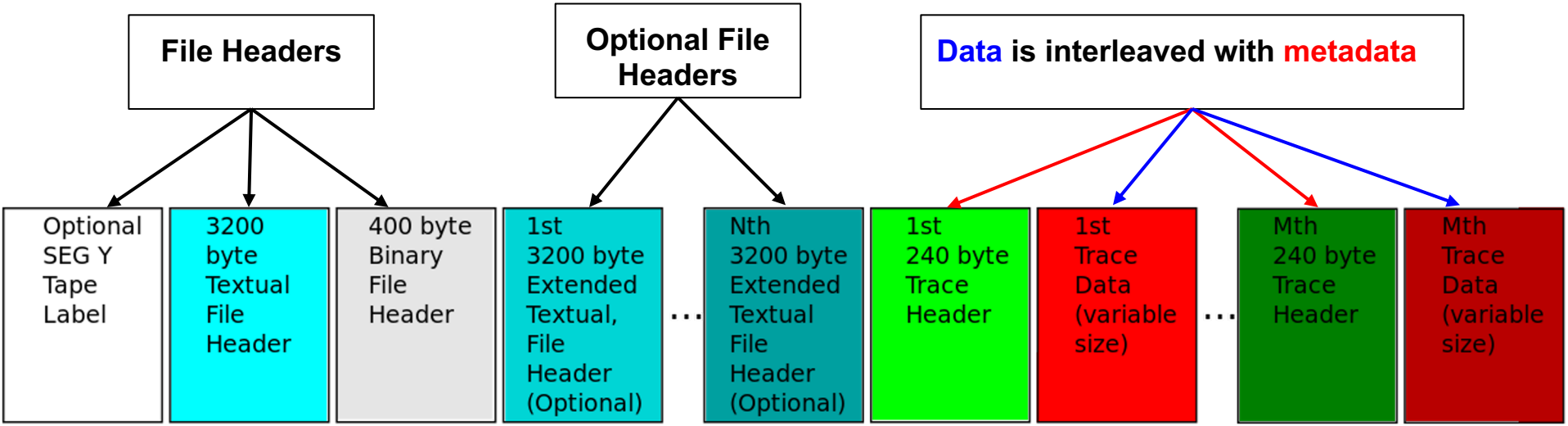
… to the final result



Agenda

- Marine seismic acquisition
- **Motivation**
- DAOS-SEIS mapping Graph
- DAOS-SEIS API
- Benchmarking Results

Seg-Y Traditional Structure



Motivation

Limitations

- A new copy of the data is created along with each processing step.
- Serial accessing of data in a segy file.
- Most processing applications access traces headers to decide the access pattern to the data.

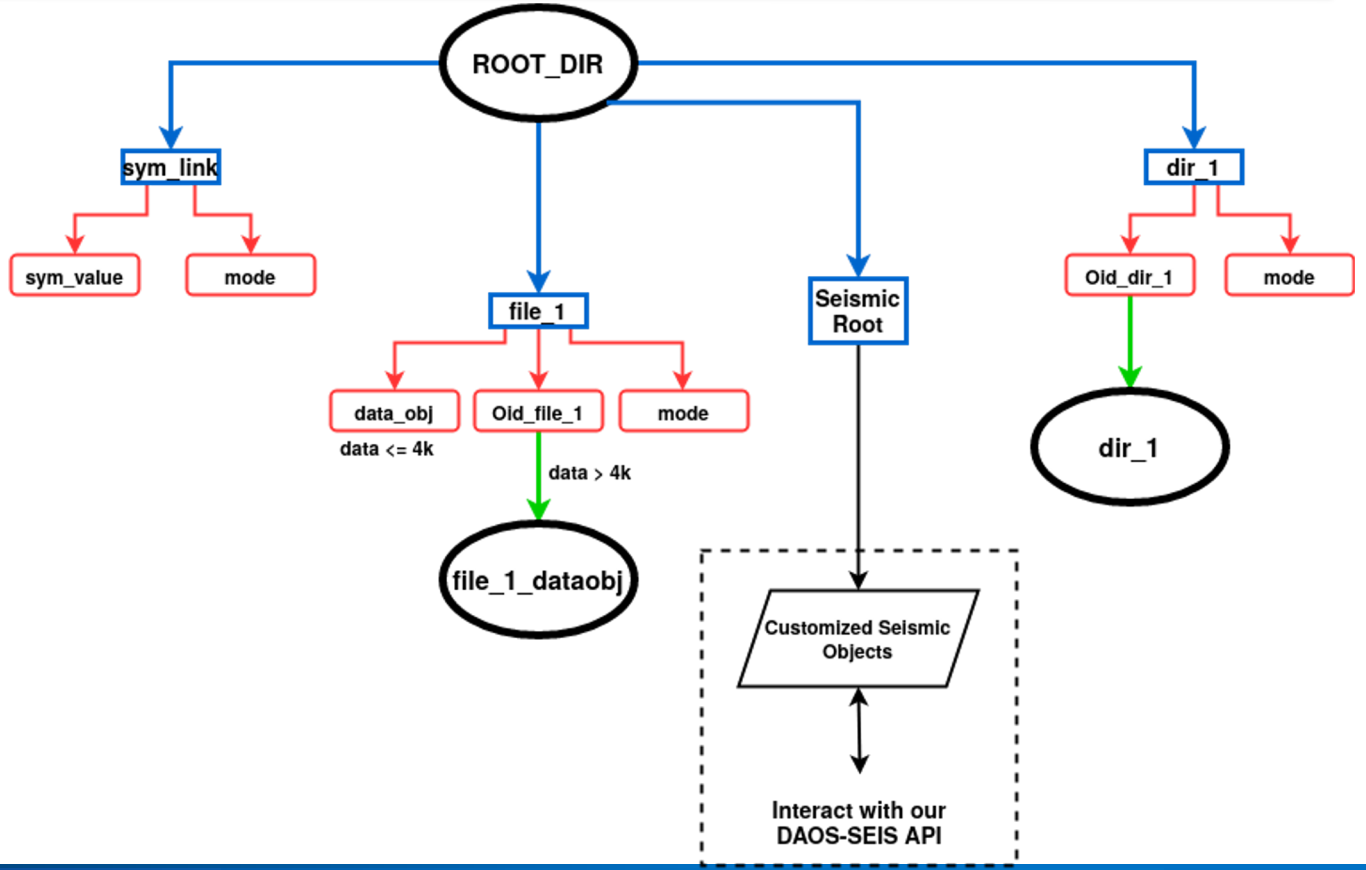
Goals

- Leverage object based storage system and inherent metadata/data separation.
- Reduce number of copies through utilizing daos snapshots.
- Store only updates through utilizing daos versioning object storage.

Agenda

- Marine seismic acquisition
- Motivation
- **DAOS-SEIS mapping Graph**
- DAOS-SEIS API
- Benchmarking Results

What is DAOS-SEIS ?

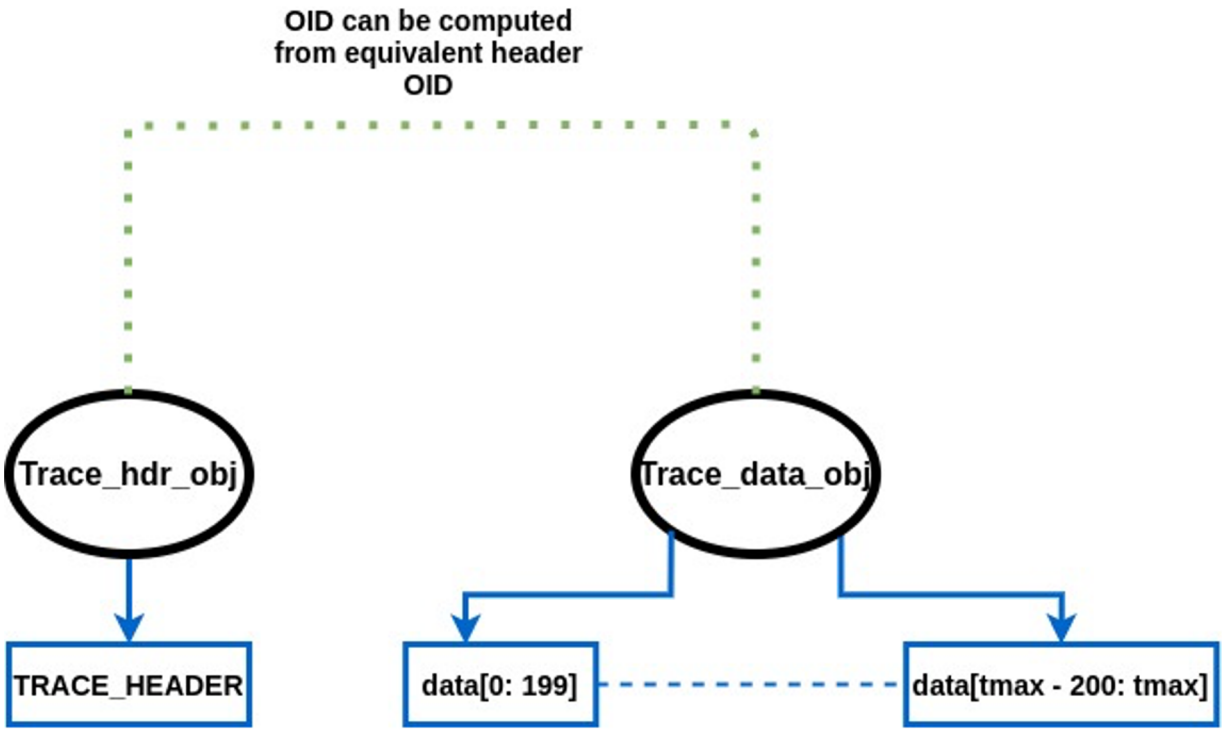


Blue : Dkey

Red : Akey

Black : Object

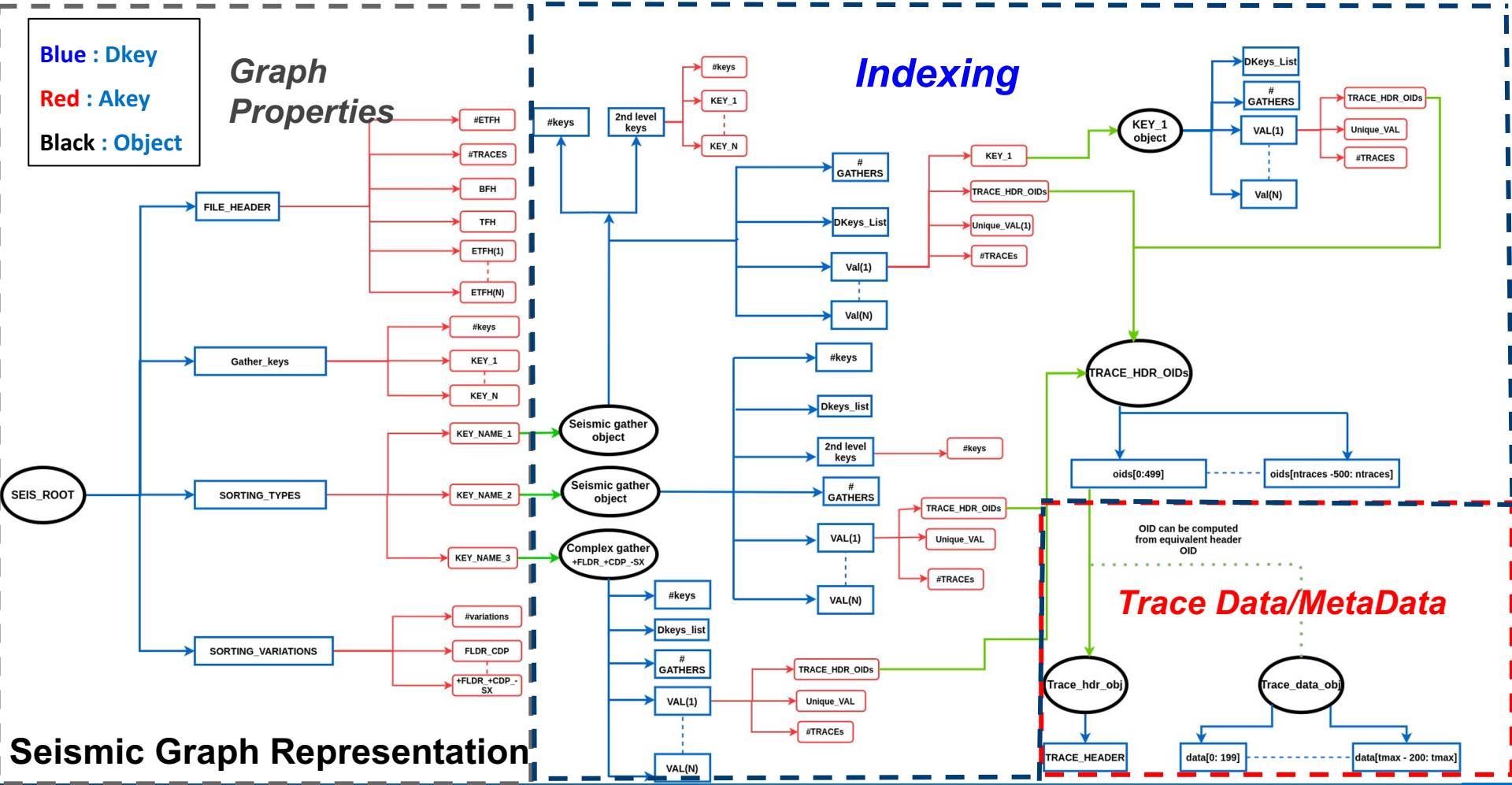
Trace Header & Trace Data



Blue : Dkey

Red : Akey

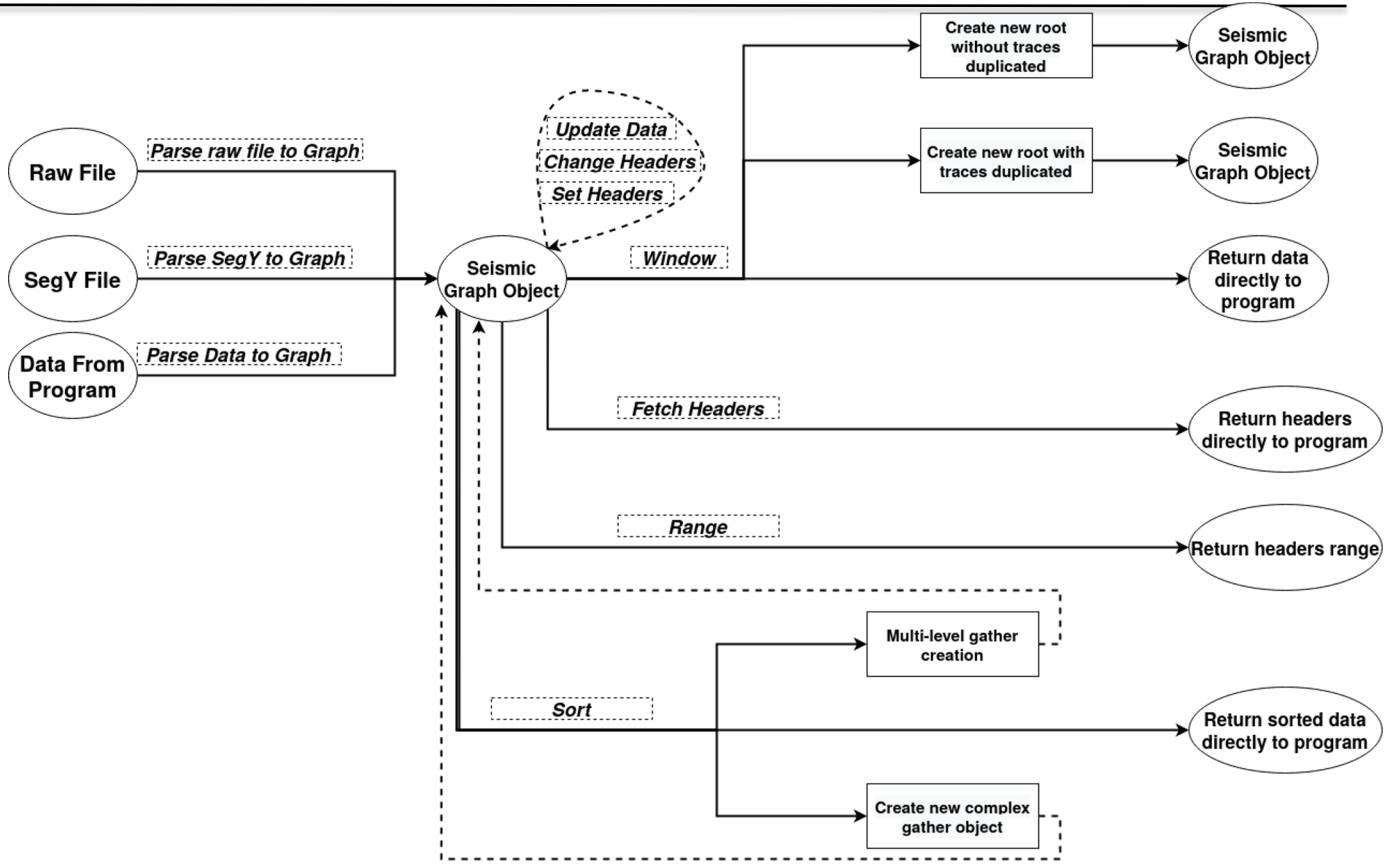
Black : Object


Blue : Dkey
Red : Akey
Black : Object
Graph Properties
Indexing
SEIS_ROOT
FILE_HEADER
#ETFH
#TRACES
BFH
TFH
ETFH(1)
ETFH(N)
Gather_keys
#keys
KEY_1
KEY_N
SORTING_TYPES
KEY_NAME_1
KEY_NAME_2
KEY_NAME_3
SORTING_VARIATIONS
#variations
FLDR_CDP
+FLDR_+CDP_-SX
Seismic gather object
Seismic gather object
Complex gather +FLDR_+CDP_-SX
2nd level keys
GATHERS
DKeys_List
Val(1)
Val(N)
Dkeys_list
Unique_VAL
#TRACEs
TRACE_HDR_OIDs
KEY_1 object
Val(N)
oids[0:499]
oids[ntraces -500: ntraces]
OID can be computed from equivalent header OID
Trace Data/MetaData
Trace_hdr_obj
Trace_data_obj
TRACE_HEADER
data[0: 199]
data[tmax - 200: tmax]
Seismic Graph Representation

Agenda

- Marine seismic acquisition
- Motivation
- DAOS-SEIS mapping Graph
- **DAOS-SEIS API**
- Benchmarking Results

DAOS-SEIS API



Agenda

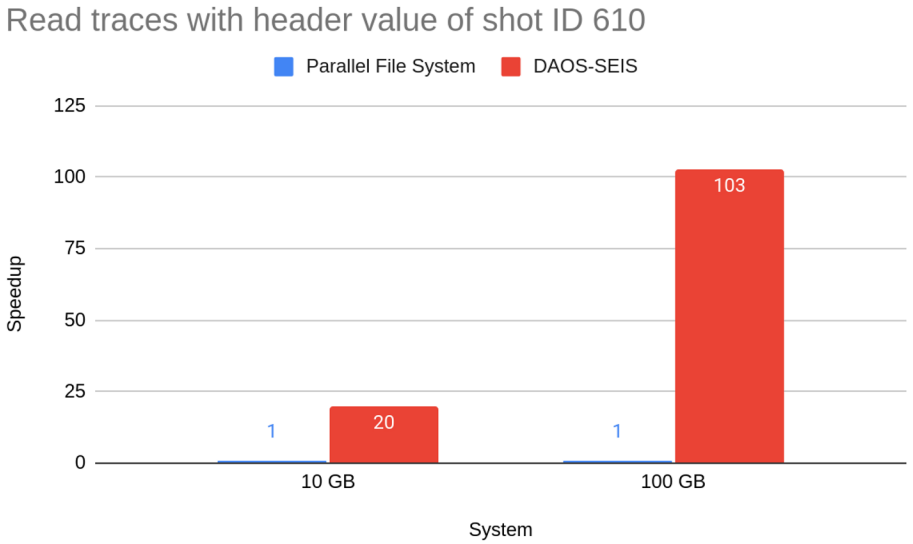
- Marine seismic acquisition
- Motivation
- DAOS-SEIS mapping Graph
- DAOS-SEIS API
- **Benchmarking Results**

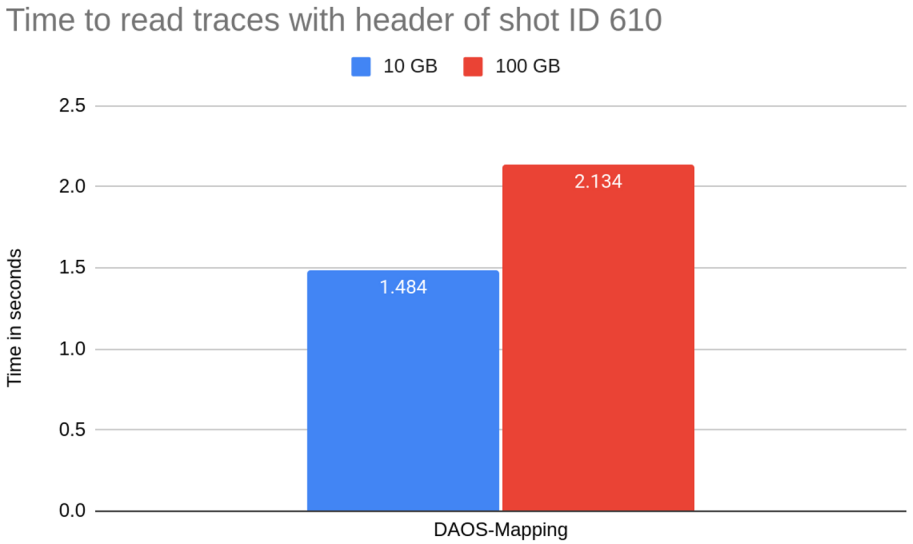
Early Results : Seismic Unix vs DAOS-SEIS

- Comparison of the well known library of seismic unix with our DAOS seismic mapping
  - Test seismic unix on a parallel file system(PFS).
  - The parallel file system is not using the same hardware used for the DAOS system.
- The daos system used to get these results is consisting 1 server node and 1 client node

Early Results : Windowing

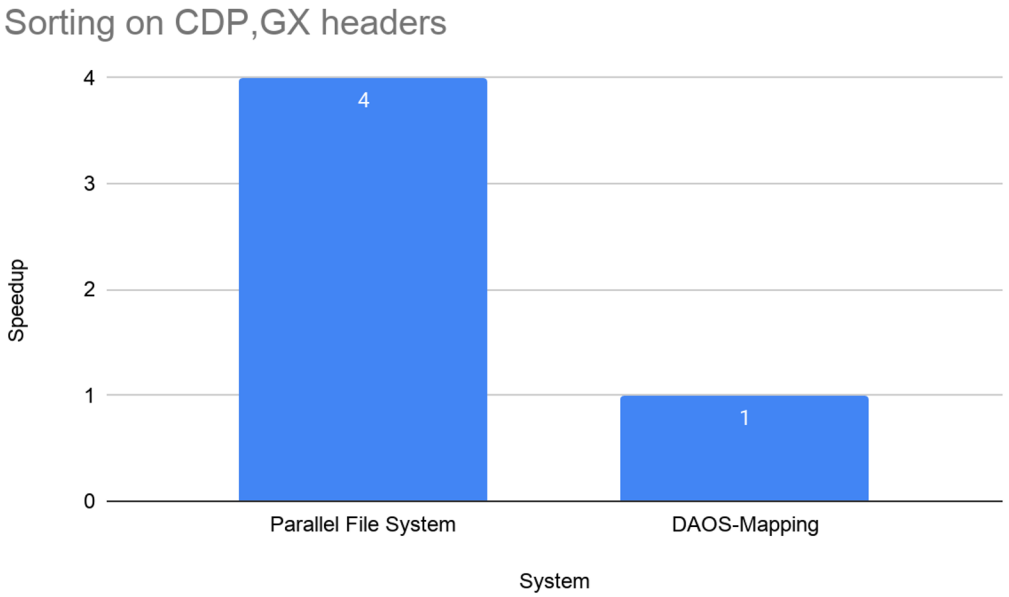
- Filtering data to read an arbitrary shot ID from the dataset.
- We can notice that the time taken for the DAOS is almost constant even when the size of the dataset increased tenfold, due to the graph we've built when parsing, just increasing from 1.5 second to 2 second.
- Speedup from seismic unix would increase with the size of data since SU time scales with the size of the data, on a 10 GB data, DAOS about 20x faster, while on 100 GB data, it was 103x faster than its equivalent.





Early Results : Sorting

- We noticed a 4x reduction on sorting from seismic unix.
- Read results slower than on PFS due to sequential read, which indicates we need to move to async API to increase queue depth.
- Dataset was also small, so we suspect a cache effect with the PFS.
- Further tuning is required.



Next Steps

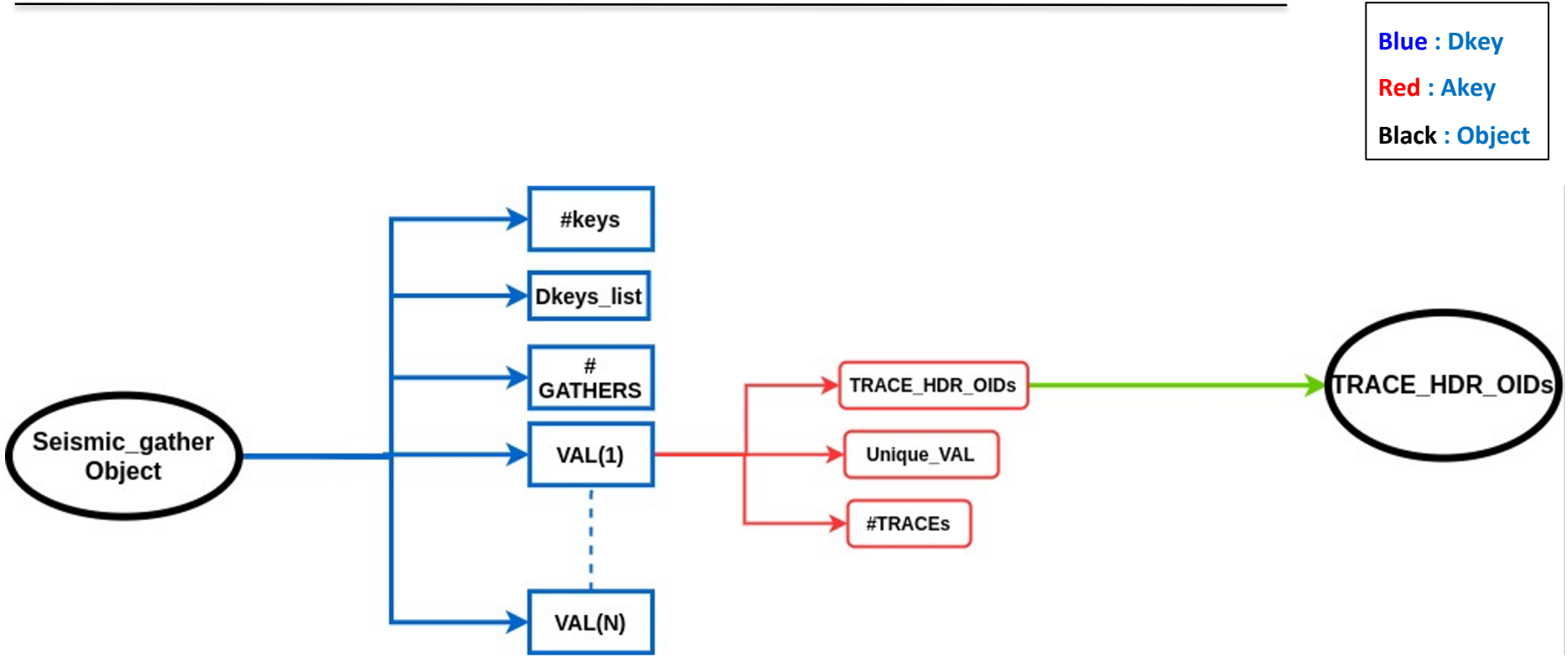
- Move to DAOS async API to increase queue depth.
- A more detailed comparison against different parallel file systems.
- Optimize current API implementation.
- Parallelize Parsing and sorting functionalities.
- Cover more Seismic processing functionalities in our API.
- Integrate DAOS-SEIS API in existing seismic processing frameworks.

Resources

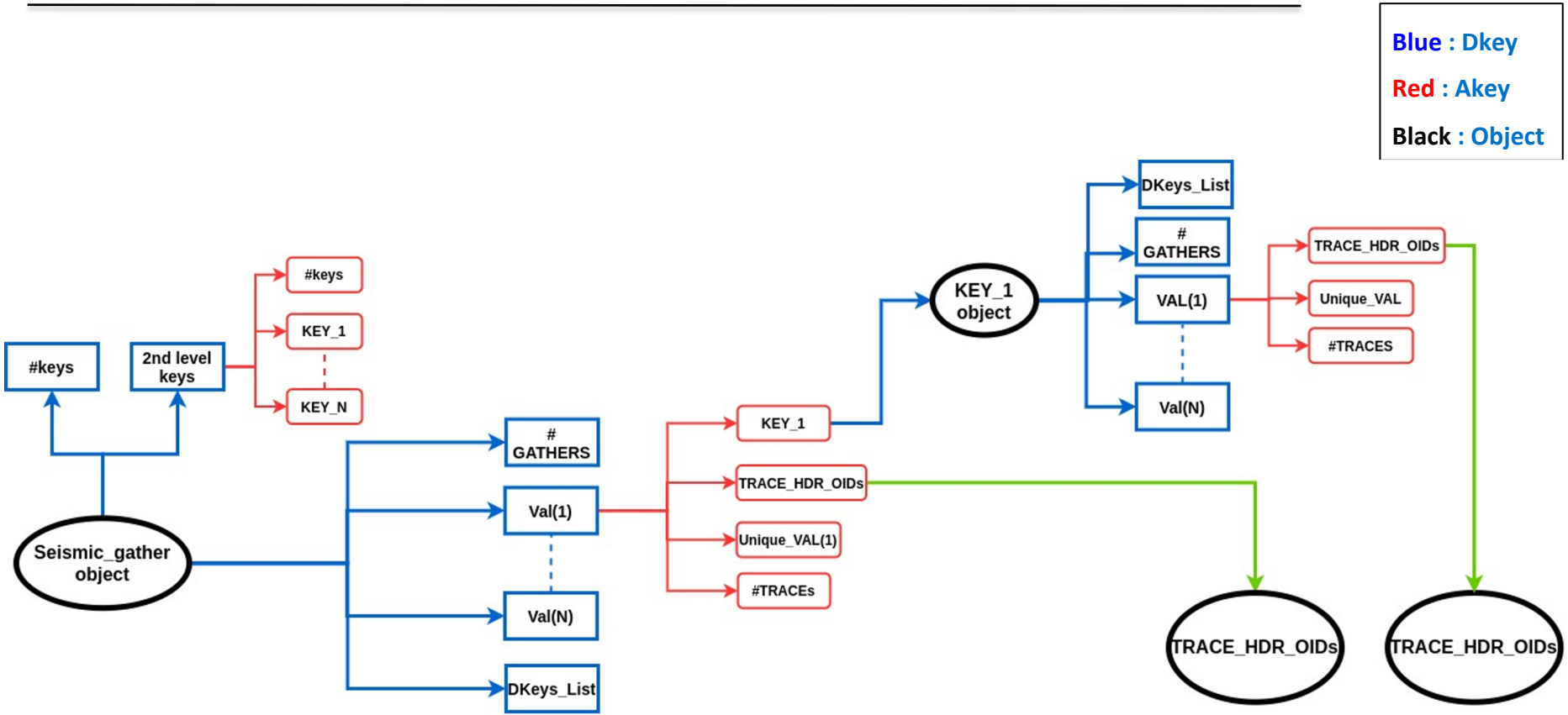
- Link to the DAOS-SEIS mapping wiki page:
  - https://wiki.hpdd.intel.com/display/DC/DAOS-SEGY+Mapping
- Link to the open-source github repository of the DAOS segy mapping:
  - https://github.com/daos-stack/segy-daos

Thanks

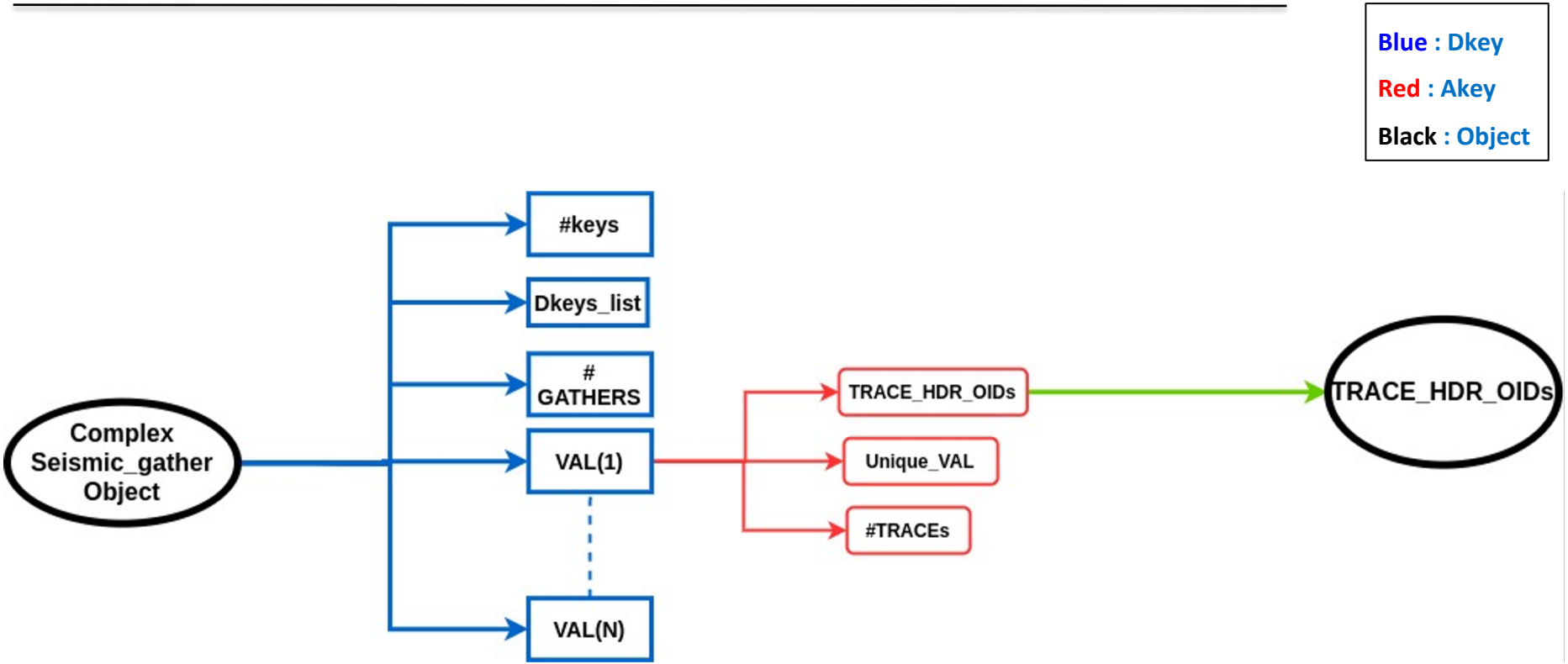
Appendix: One level seismic gather object



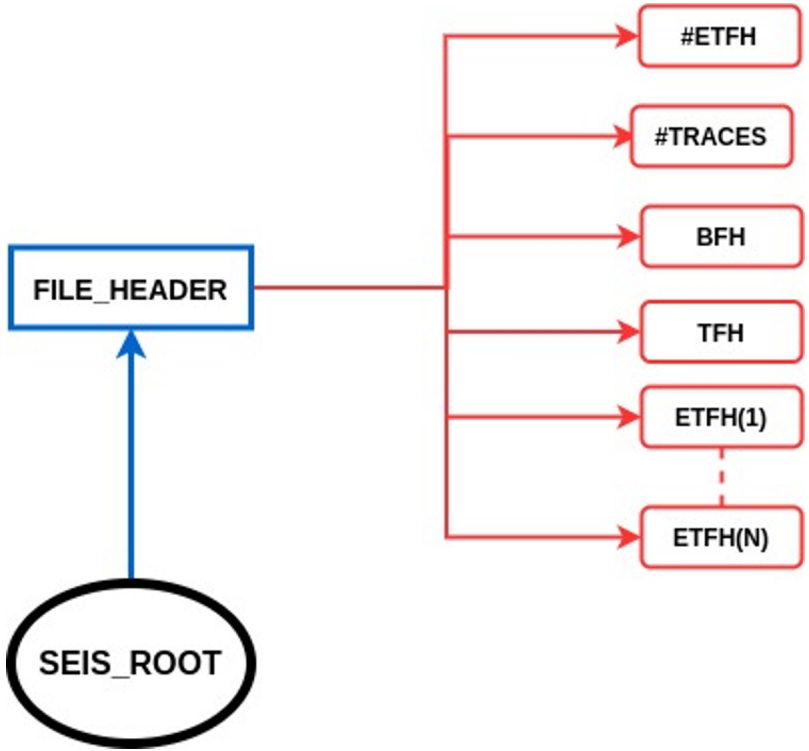
Appendix: Multi-Level seismic gather object



Appendix: Complex gather object



Appendix: File Header

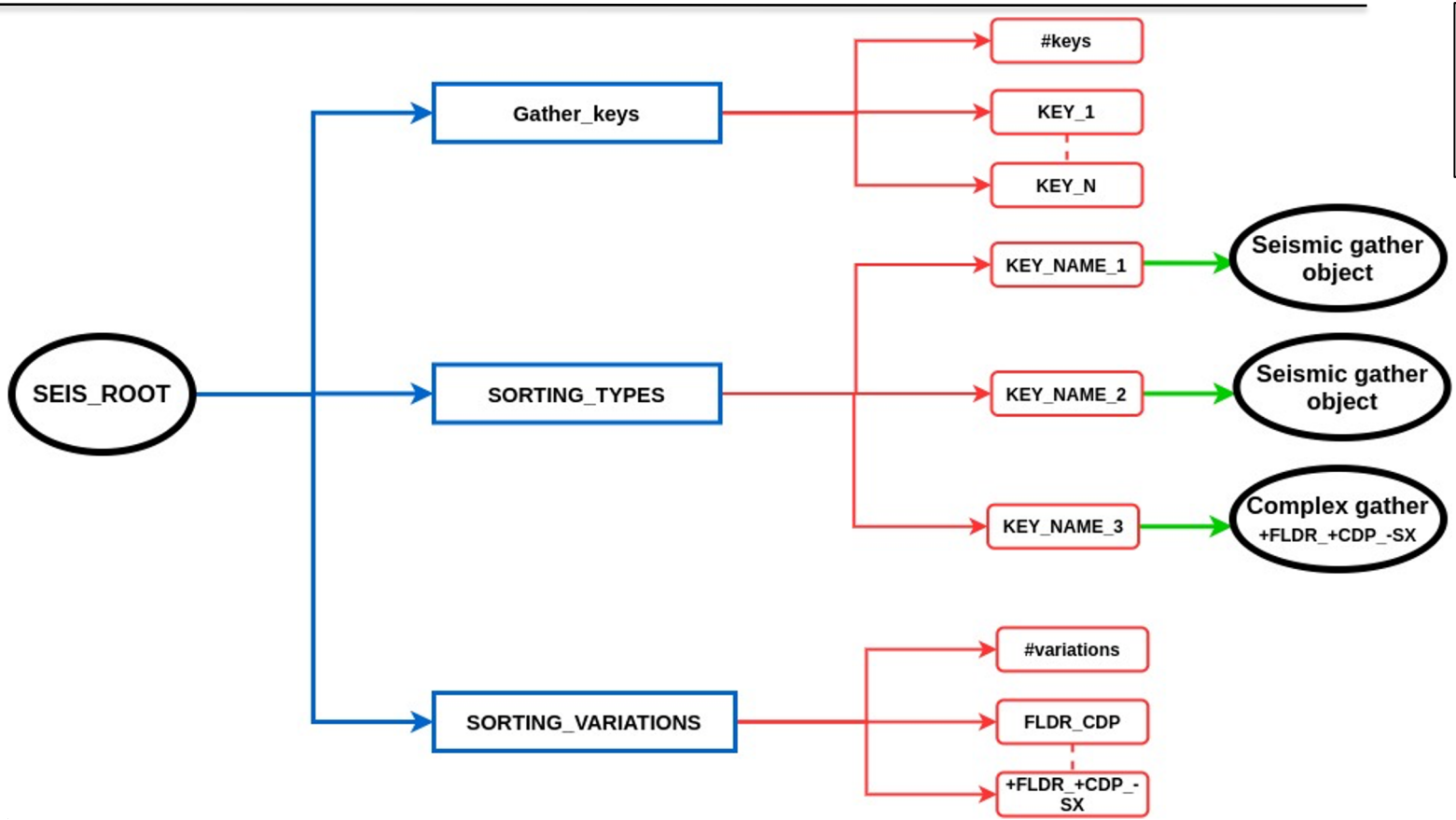


Blue : Dkey

Red : Akey

Black : Object

Appendix: Root Seismic Object



Appendix: Window functionality