KaraboHDF5 Storing Scientific Data at the European XFEL



Jose Luis Vázquez-García Data Department Controls Software Engineer

Hamburg, May 26, 2025

The European XFEL

X-ray Free Electron Laser

3 SASE¹ lines, 7 instruments

Characteristics

High volume of generated data (~PB/day)

High speed of data generation (~15 GB/s)





The European XFEL

X-ray Free Electron Laser

3 SASE¹ lines, 7 instruments

Characteristics

High volume of generated data (~PB/day)High speed of data generation (~15 GB/s)

Problems:

Storage hardware cannot be expanded forever.

Both hardware and software must be able to handle high speed of data generation

```
<sup>1</sup> Self-amplified Spontaneous
Emission European XFEL
```

Jose Luis Vázquez-García, Data Department Group, May 26, 2025



Raw Data Generated at European XFEL Instruments



SCADA system, written from scratch in C++ and Python.

Event-driven, fault-tolerant, highly distributed.

Modular:

- *Karabo devices* make up the system.
- Karabo Hash: flexible data structure to store and communicate data between devices.



The Hash:

- De facto karabo standard data structure to store and transmit data.
- **Flexible hierarchical** container.
- *Keyword value* container.
- Values can be almost any type, including other hashes.
- Attributes (keyword-value pairs) can be attached to values.
 - Provides a tree-like interface to data.

In [1]: from karabo.middlelayer import Hash
In [2]: h = Hash('detector', 'AGIPD',

...: 'geometry', [2, 128, 256])

In [3]: print(h)
<detector{}: 'AGIPD', geometry{}: [2, 128, 256]>

The Hash:

- De facto karabo standard data structure to store and transmit data.
- **Flexible hierarchical** container.
- *Keyword value* container.
- Values can be almost any type, including other hashes.
- Attributes (keyword-value pairs) can be attached to values.
 - Provides a tree-like interface to data.

The Hash:

- De facto karabo standard data structure to store and transmit data.
- **Flexible hierarchical** container.
- *Keyword value* container.
- Values can be almost any type, including other hashes.
- Attributes (keyword-value pairs) can be attached to values.
 - Provides a tree-like interface to data.

```
In [1]: from karabo.middlelayer import Hash
In [2]: h = Hash('detector', 'AGIPD',
                  'geometry', [2, 128, 256])
In [3]: print(h)
<detector{}: 'AGIPD', geometry{}: [2, 128, 256]>
In [4]: attributes = {'timeStamp': 12345,
                       'version': '1.0'}
In [5]: value = Hash('language', 'Python',
                     'version', '3.12')
 In [6]: h.setElement(key = 'api',
                     value = value,
                     attrs = attributes)
 n [7]: print(h)
<detector{}: 'AGIPD', geometry{}: [2, 128, 256], api{'timeStamp': 12345, 'version': '1.0'}: <</pre>
language{}: 'Python' => String
version{}: '3.12' => String
```

The Hash:

- De facto karabo standard data structure to store and transmit data.
- **Flexible hierarchical** container.
- *Keyword value* container.
- Values can be almost any type, including other hashes.
- Attributes (keyword-value pairs) can be attached to values.
 - Provides a tree-like interface to data.

```
In [1]: from karabo.middlelayer import Hash
In [2]: h = Hash('detector', 'AGIPD',
                 'geometry', [2, 128, 256])
In [3]: print(h)
<detector{}: 'AGIPD', geometry{}: [2, 128, 256]>
In [4]: attributes = {'timeStamp': 12345,
                       'version': '1.0'}
In [5]: value = Hash('language', 'Python',
                     'version', '3.12')
In [6]: h.setElement(key = 'api',
                     value = value,
                     attrs = attributes)
[n [7]: print(h)
<detector{}: 'AGIPD', geometry{}: [2, 128, 256], api{'timeStamp': 12345, 'version': '1.0'}: <</pre>
language{}: 'Python' => String
version{}: '3.12' => String
>>
In [8]: print(h['detector'])
AGIPD
In [9]: print(h['api'])
<language{}: 'Python', version{}: '3.12'>
In [10]: print(h['api.language'])
Python
```

The control system: Karabo (iii)

HED_DAQ_RUN_MGMT - European XFEL - Karabo GUI 2.21.0 - Topic: HED - User: vazquezj								- a :	
Eile Links View Iools Help									
🔀 🖲 🔒 🎯									
System Topology Device Topology Projects	HED_RUN_CONTROL X					Configuration Editor			
🗵 🛎 🔶 🔶	♦ I ✓ X								B ≞ ⊕ ÷
Find: Filter Clear Filter	HED Run Controller	HED_DAQ_DATA/DM/MONITOR	Check DAO	Ok	HED_DAQ_DA	Find:			Filter Clear Filter
No Status Filtering	Data Source Groups		Group to Dat	ta Aggregator N	lapping	Property	Current value on device	Value	
Service Manager		Filter Clear Filter	Data Source	Alias		A _DeviceID_	HED_IA1_JF500K5/CA		
Host - Server - Class - Device	Selected Group	^	0 @EXP_SAM	DAOI	HED_DAQ	A DeviceID	HED_IA1_JF500K5/CA		
G v √ cppHED/jungfrau5	► V HED_LT_CAM		1 @HED_LT_CAM	DA10	HED_DAQ	Heartbeat interval	120		
 JungfrauControl 	► V HED_PPL		2 @HED_PPL	DA11	HED_DAQ	A ServerID	middlelayerServer/he		
HED_IA1_JF500K5/DET/CONTROL	5438_CAMS								
🥐 👻 🖌 pythonServer/hed_jungfrau5	→ 5438_SLITS					Visibility	U		
 ImageAverager 	▶ 5438_STAGES					ClassID	OfflineCalRunner		
HED_IAI_JF500K5/AVG/RECEIVER	ACA2440_CAMS					6 Class version	OfflineCalRunner-1.0.3		
HED IA1 IESONS/RSUB/RECEIVER	AGIPD65K1_CTRL AGIPD65K1_TSYS						onanceandanier 1.0.5		
 ImageProcessor 	AGIPD65K1_ISIS					A Karabo version	2.20.8		
HED_IA1_JF500K5/PROC/RECEIVER	AGIPD500K2G_CTRL_PWR					A ServerID	middlelayerServer/he		
👻 💻 hed-br-kc-comp-1	AGIPD500K2G_LITFRM								
niddlelayerServer/hed_onc_cal	AGIPD500K2G_TSYS					A Host	hed-br-kc-comp-1		
 OfflineCalRunner 	AGIPDHZ0_XTDF					1 Process ID	2127358		
HED_EXP_AGIPD500K2G/CAL/RUNNER	AGIPDHZ1_XTDF					State	DASSIVE		
HED_IA1_JF500K1/CAL/RUNNER	AGIPDHZ2_XTDF					June	TASSIVE		
HED_IA1_IF500K3/CAL/RUNNER	AGIPDHZ3_ATDF					A Status			
HED IA1 JF500K4/CAL/RUNNER	AGIPDHZ_CTRL					E Alarm condition	none		
HED_IA1_JF500K5/CAL/RUNNER	AGIPDHZ_POWER								
niddlelayerServer/hed_onc_misc	AGIPDHZ_TSYS					Locked by			
🕐 👻 🖌 middlelayerServer/hed_onc_scan	CAMS 5438					Clear Lock			
 SimpleScanner 	CNRS_SPEC					A Last command			
HED_EXP_SCAN/DA/SIMPLE_SCANNER	COM_LAS_CAM								
 MiddlelayerServer/runvalidator Buol/alidator 		*				Logger			
HED DAO DA/RUN VALIDATOR/1						Device interfaces	['Processor']		
pythonServer/hed_zma01	Preparation							1	
🐣 👻 🖌 pythonServer/metroProcessor	7004	(16:35:09): Run folder close	a successfully			A Processing node	xcal@sa2-onc-01	xcal@sa2-onc-01	
 MetroBound 	Proposal number 7904 7904					Virtual environment	/scratch/xcal/pycalibr	/scratch/xcal/pycalibration/.venv/	
HED_EXP_DA/METRO/1	Proposal instrument HED					A BunController device ID			
HED_EXP_DA/METRO/SPEC		1					11ED_DAQ_DATA/DM/		
HED_EXP_DA/METRO/VISAR	Proposal title Resolving the valence electronic structure of iron at solar interior conditions					E Detector	JUNGFRAU	JUNGFRAU	
 med-br-kc-comp-2 middlelawerSener/hed_arind500k_lifframe_finder 	Data and the second sec	-				Action	CORRECT	CORRECT	
 Aripdi itFrameFinder 	/gpts/extel/exp/HED/202501/p007904/raw	J						The second	
HED EXP AGIPD500K2G/REDU/LITFRM						Arguments		Table Element	
🥐 🖌 pythonServer/agipd_adu	Retrieve proposal Push to DAQ Apply configuration Monitor data					1 Proposal	5438	5438	
pythonServer/hed_zmq02	Clear configuration Ignore data	L				Bun	20	20	
Hed-br-sys-cal-1					6				
n with the second secon	Setup Acquisition Cancel Setup					Output Directory	proc	proc	
AgipdCondition						Use previous constants	False	False	
Arind500KGeometry						Concentration former			
HED_DET_AGIPD500K2G/CAL/GEOMETRY						same constants as for run			
 CalibrationManager 						Last shell command executed to perform a correction			
HED_DET_AGIPD500K2G/CAL/MANAGER						DAD readiness timeout	10 s	10 s	
👻 🗢 RoiTool									
HED_DET_AGIPD500K2G/CAL/ROI_HISTOGRAM_CORR						ZMQ publish port	10347		
v v pyhŁu/cai_agipd_extra						Process			
HED DET AGIPD500K2G/CAL/PICKED PREVIEW						Start monitor	1		
 SaturationWarningAggregator 						Start monitor			
HED_DET_AGIPD500K2G/CAL/SATURATION_WARNING_AGGREGATOR	4				Þ	Shutdown instance Apply all	Decline all		
025-05-22 15:01:06 - INFO - Executing slot requestScene of instance MID_EXP_AGIPD1M/PSC/HV 025-05-23 10:23:16 - INFO - Disconnected from the gui server localhost:4446 025:06:23 10:23:19 - INFO - Successfully connected to gui server (loci): hed-rs-ys-con-gui:4444 (HED) 025:06:24 10:27:24: INFO - Joadine oncient HED JOA DB UN GMUT from morited database								hed-rr-sys-con-	gui1:44444 (HED)

The control system: Karabo (iii)



Data sources

AGIPD

MSSC

etc...





AGIPD Imager (front) and captured image shown in Karabo (back)

European XFEL

Data Generation (iii)

Many data formats must be handled



format

Data Reduction: *when* storing the data

Data compression

- Preliminary tests with data compressed with zlib + HDF5 did not provide enough storage speed.
- HDF5 compression + chunking fits the requirements:
 - On acquisition start, create a file template with chunks along the 3rd axis (pulses).
 - Chunks with no data will not be written.



Data Reduction: when storing the data

Data compression

- Preliminary tests with data compressed with zlib + HDF5 did not provide enough storage speed.
- HDF5 compression + chunking fits the requirements:
 - On acquisition start, create a file template with chunks along the 3rd axis (pulses).
 - Chunks with no data will not be written.



The DAQ (i)

Data Acquisition System: **Detectors + Hardware + Karabo** devices

Data Aggregators:

- A Data Aggregator is a *Karabo device*
- Each aggregator gathers data from a given set of sources.
- Stores its data in one HDF5 file (one file per aggregator).
- Real-time data recording is possible, by storing data in several HDF5 files in parallel.

The DAQ can be scalated, by adding more aggregators.



The DAQ (ii)

Sustained writing rates to **gpfs** storage over 15 GB/s



17

KaraboHDF5 (i)

C++ library written on top of the HDF5 C API.

Maps Hash data structures to HDF5 data objects.

A Hash can be mapped to a Group.

- A *value* can be mapped to a *Dataset*.
- Attributes of a Hash entry map to HDF5 attributes of a Dataset.
- karaboHDF5 uses compression and data chunking to optimise storage.



KaraboHDF5 (ii)



KaraboHDF5 (ii)

0-0	Start	'boolProp	perty' => 0 B00L
0-0	Stop	'floatPro	operty' => 215.0398 FLOAT
0-0	Reset	'node' +	Toperty => 22 INTS2
1	Vector size	'float	Property' => 883.1215 FLOAT
\checkmark	Bool property	'vectorPı 27897,262	<pre>roperty' => 22677,28972,25975,10028,31406,11311,8681,11494,1190,10879, 222,32179,30045,13779,27906,14467,17921,31452,18375,24475,1427,276,305</pre>
1	Integer property	99,3010,2	26499,8619,936,3456,20927,28255,6061,23382,30380,29470,2399,2594,30286
1.1	Float property	211 211 211 217 217 217 217 217 217 217	3,28922,12844,480,31222,6921,7667,28922,22383,20249,3059,756,16720,171
	Vector property	[22677 0, 1, 1, 1, 1, 1	L,1,1,0,1,1,0,0,1,1,0,1,1,0,0,1,0,1,0,1,
	Vector boolean property	[True F'trueFree	uency' => 9.999718 FLOAT
•	Node Values		
	🔝 Node Float property	883.1215 mV	
-	Output		
	📃 Unused, was Distri	load-balanced	
	📃 No Input (Shared)	drop	
	I llestname	default	

KaraboHDF5 (ii)	~ RUN ~ XHO_EG_DG ~ DATA	
Image: Start Image: Stop Image: Stop	<pre>'boolProperty' => 0 B00L 'floatProperty' => 215.0398 FL0AT 'integerProperty' => 22 INT32 'node' + 'floatProperty' => 883.1215 FL0AT 'vectorProperty' => 22677,28972,25975,1 27897,26222,32179,30045,13779,27906,144 99,3010,26499,8619,936,3456,20927,28255 857,24713,28922,12844,480,31222,6921,76 'vectorBoolProperty' => 1,0,0,1,0,0,0,0 [22677 0,1,1,1,1,1,1,0,1,1,0,0,1,1,0,1,1,0,0,1 TrueF'trueFrequency' => 9.999718 FL0AT 883.1215 mV load-balanced</pre>	<pre>> 1 > alarmCondition > autostart > availableScenes > boolProperty > classId > classVersion > deviceId > floatProperty > hostName > integerProperty > karaboVersion > lastCommand > lockedBy > node > seed > serverId > speedUp > state</pre>	94,1190,10879, 5,1427,276,305 399,2594,30286 ,756,16720,171 0,0,1,0,1,1,1, 0 VECTOR_BOOL
No Input (Shared)	drop	 > status > trueFrequency > updateFrequency 	
Europear	XFEL	 vectorBoolProperty vectorProperty vectorSize 	

Data Reduction: before storing the data

Sheer amount of data is a problem: let's try to reduce it before saving the data.

Pulse-based reduction.

Ignore images for pulses that have no information.

Train-based reduction.

Ignore trains for pulses that have no information.



Data Reduction: before storing the data

Sheer amount of data is a problem: let's try to reduce it before saving the data

Pulse-based reduction.

Ignore images for pulses that have no information.

Train-based reduction.

Ignore trains for pulses that have no information.

Gain compression, for AGIPD detectors.

Each pixel can have one of three gain stages, depending on incident number of photons.

In case of fixed gain, the gain array can be discarded.

