EIGER2 data interfaces

Max Burian | Product Manager Synchrotron

Diego Gämperle | Product Owner

DECTRIS
detecting the future

HDRMX contacts @ DECTRIS

Diego Gämperle - Product Owner

Team Lead – Software Development
Contact: diego.gaemperle@dectris.com

Max Burian – Product Manager Synchrotron

Synchrotron & Application Specifics
Contact: max.burian@dectris.com

DECTRIS
detecting the future

Data Interfaces

– *Data interfaces enable access to image series*

– *Multiple data interfaces are available*

  – *different beamlines have different requirements*

– *Multiple data interfaces can be active simultaneously*

  – *no more than 2 high performance interfaces active at any given time*

DECTRIS
detecting the future
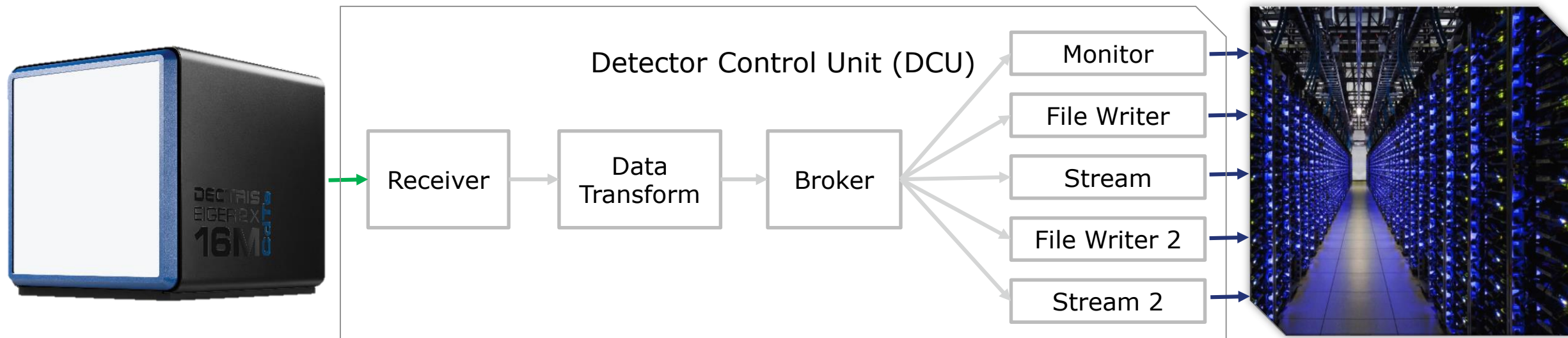
Stream 2

- *ZeroMQ*
- *Clients can define amount of metadata they receive*
- *Compressed metadata*
- *All thresholds will be sent to the same server to allow post-processing*
- *CBOR as data format (superset of JSON)*
- *Supports multi-threshold data*

2020-08-14
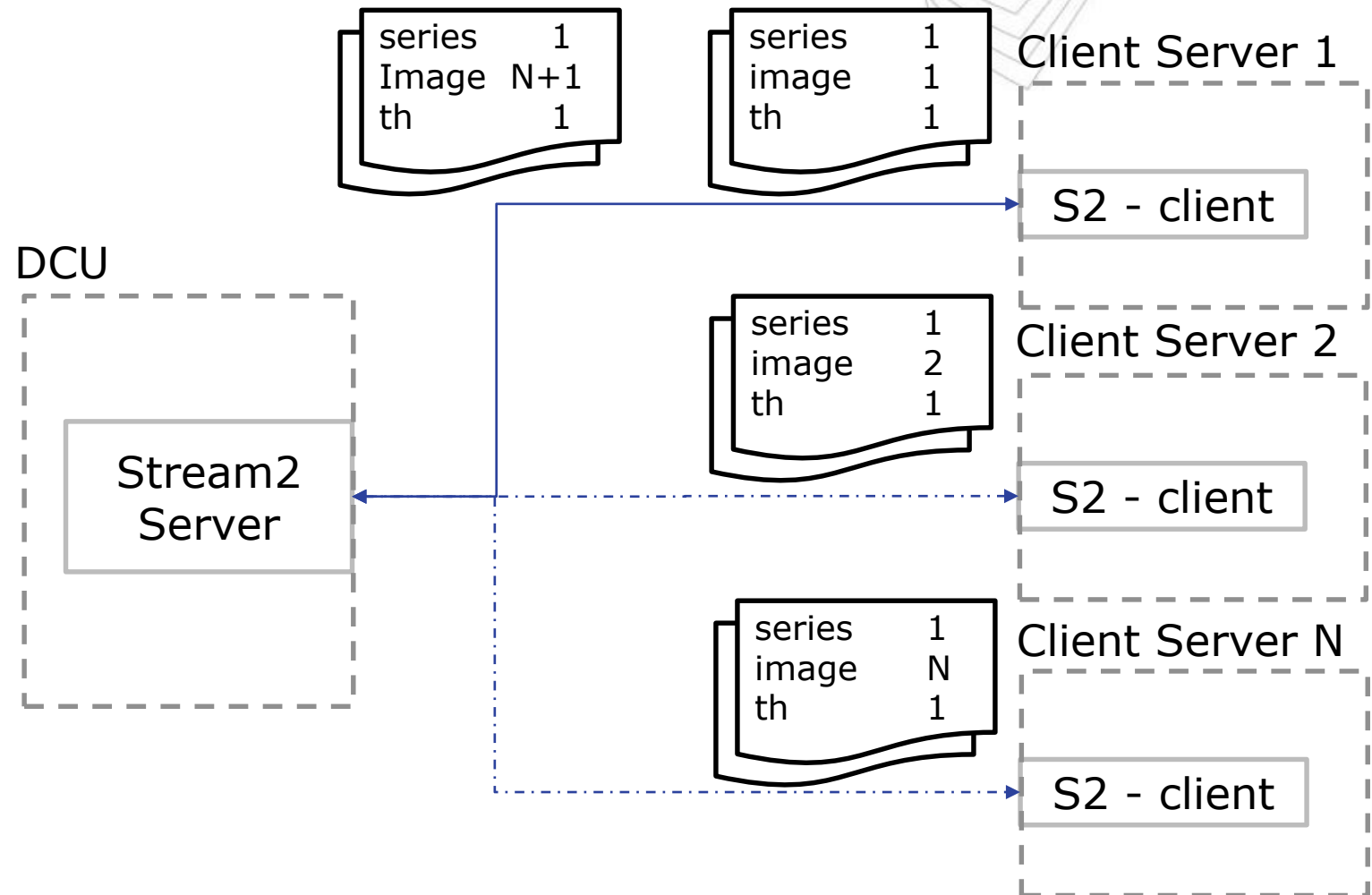
Stream 2 – current state

– *First draft has been implemented*

  – *We will begin contacting a few selected people for alpha testing soon*

– *Documentation and examples available on github*

  – *Examples for python (simplistic) and C*

Start Message

This message is sent once at the start of a series during the arm command. It contains the relevant series configurat

Field	Type	Description
type	TextString	The value "start"
beam_center_x	Float	x-position where the direct beam would hi
beam_center_y	Float	y-position where the direct beam would hi
channels	Array(Map)	Information about each data channel
data_type	TextString	The type of the channel data
thresholds	Array(Integer)	Threshold identifier(s) of the channel. For c two threshold IDs.

Image Message

This message is sent for every exposure. Each message contains all active channel (threshold) data of the exposure.

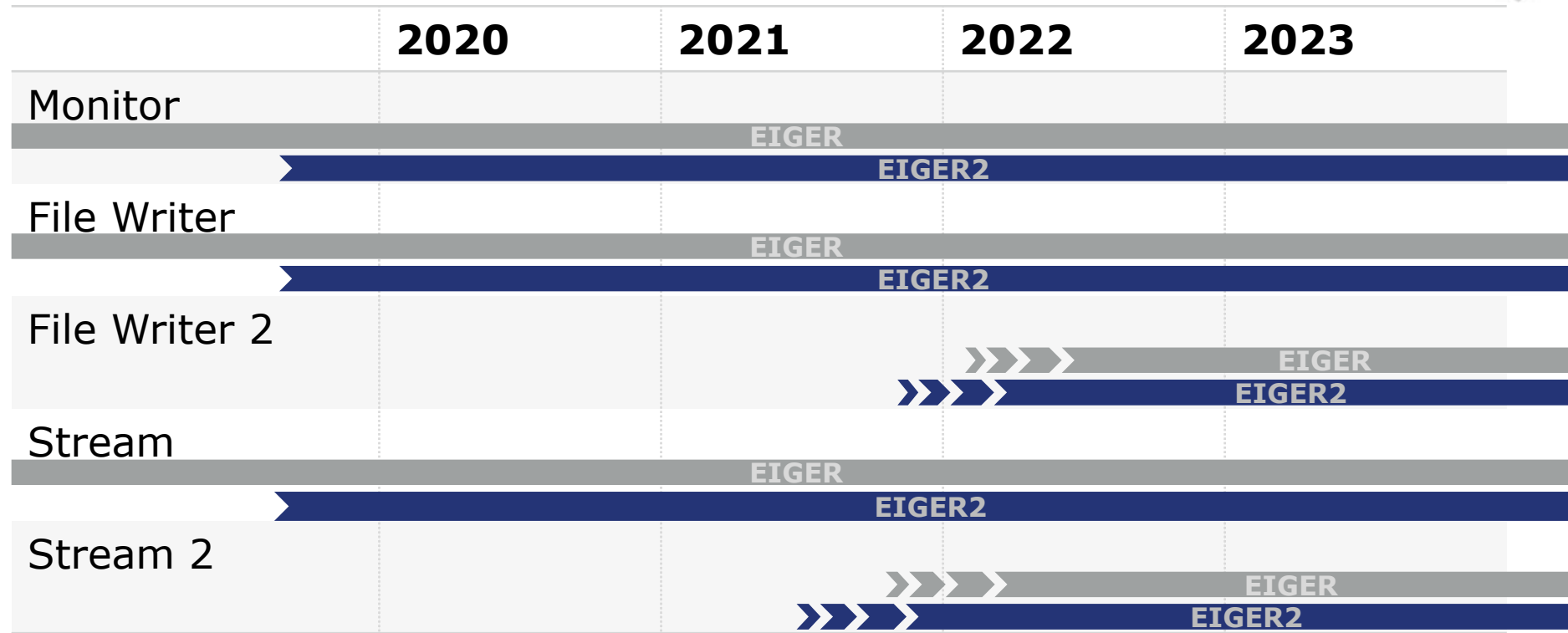| Field	Type	Description
type	TextString	The value "image"
channels	Array(Map)	The sequence of data channels
compression	TextString	Compression type (e.g. "bslz4" or "lz4")
data_type	TextString	Type of the channel data
lost_pixel_count	Integer	Count of lost pixels
thresholds	Array(Integer)	Threshold identifier(s) of the channel. For difference mode there are two threshold IDs.
data	MultiDimArray (untagged)	Channel data (compressed)
hardware_exposure_time	Array(Integer)	Hardware exposure time of the image
hardware_start_time	Array(Integer)	Hardware start time of the image
hardware_stop_time	Array(Integer)	Hardware stop time of the image
image_number	Integer	The image number
series_number	Integer	The series number
series_unique_id	TextString	The unique ID of the series

DECTRIS
detecting the future

File Writer 2 – current state

– *Started with getting to know the gold standard and it's differences*

– *Latest (at release date) HDF5 version will be used*

– *Currently unclear how to handle high speed issues*

  – *Virtual data sets*

  – *Implement solution for writing in parallel (likely OS)*

– *Few points are currently unclear, e.g:*

  – *Best way to clarify open questions (github nexus definition?)*

  – *Multi-threshold representation ([n,ch,y,x], ch = "threshold")*

  – *Handling diff images – no clear link to threshold or any other information (for example no flatfield)*

2020-08-14

DECTRIS
detecting the future

File Writer 2 – «fallout»

– *ALBULA2 becomes a necessity (gold standard support)*

  – *Complete rewriter of ALBULA*
  – *Currently in planning*
  – *Will be open source*

– *Neggia*

  – *Filewriter 1 and 2 standards will be supported (file writer 1 + 2)*
  – *recently updated to be compatible with EIGER1 and EIGER2 HDF5 files*
  – *https://github.com/dectris/neggia*

DECTRIS
detecting the future

Overview

	2020	2021	2022	2023

Monitor
EIGER
EIGER2

File Writer
EIGER
EIGER2

File Writer 2
EIGER
EIGER2

Stream
EIGER
EIGER2

Stream 2
EIGER
EIGER2

EIGER
EIGER2

DECTRIS
detecting the future