Current and planned technical solutions at ESRF for high-throughput data acquisition and data management

Pablo Fajardo on behalf of many ESRF staff with special contributions from **Andy Götz, Laurent Claustre, Alejandro Homs, Armando Solé, Fernando Calvelo, Christian Nemoz and Wassim Mansour**

ESRF | The European Synchrotron

OUTLINE

- **Overview of data acquisition and management at ESRF**
  - **Data acquisition, storage schemes and data analysis**
  - **ESRF data policy and management strategy**
- **Developments for high-throughput data acquisition**
  - **Distributed LIMA library**
  - **RDMA based framework (RASHPA)**

AN ESRF STRATEGY FOR DATA ACQUISITION ?

The current situation is the result of a combination of

- Various initiatives and action taken by various groups in different times
- There are no a fully homogeneous schemes

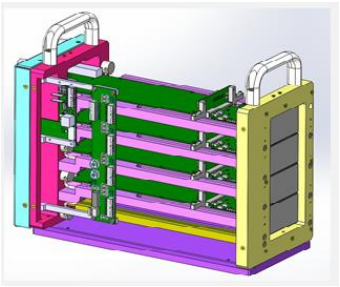
High data throughput detectors @ ESRF

Commercial

pco.edge

> 1 GByte/s

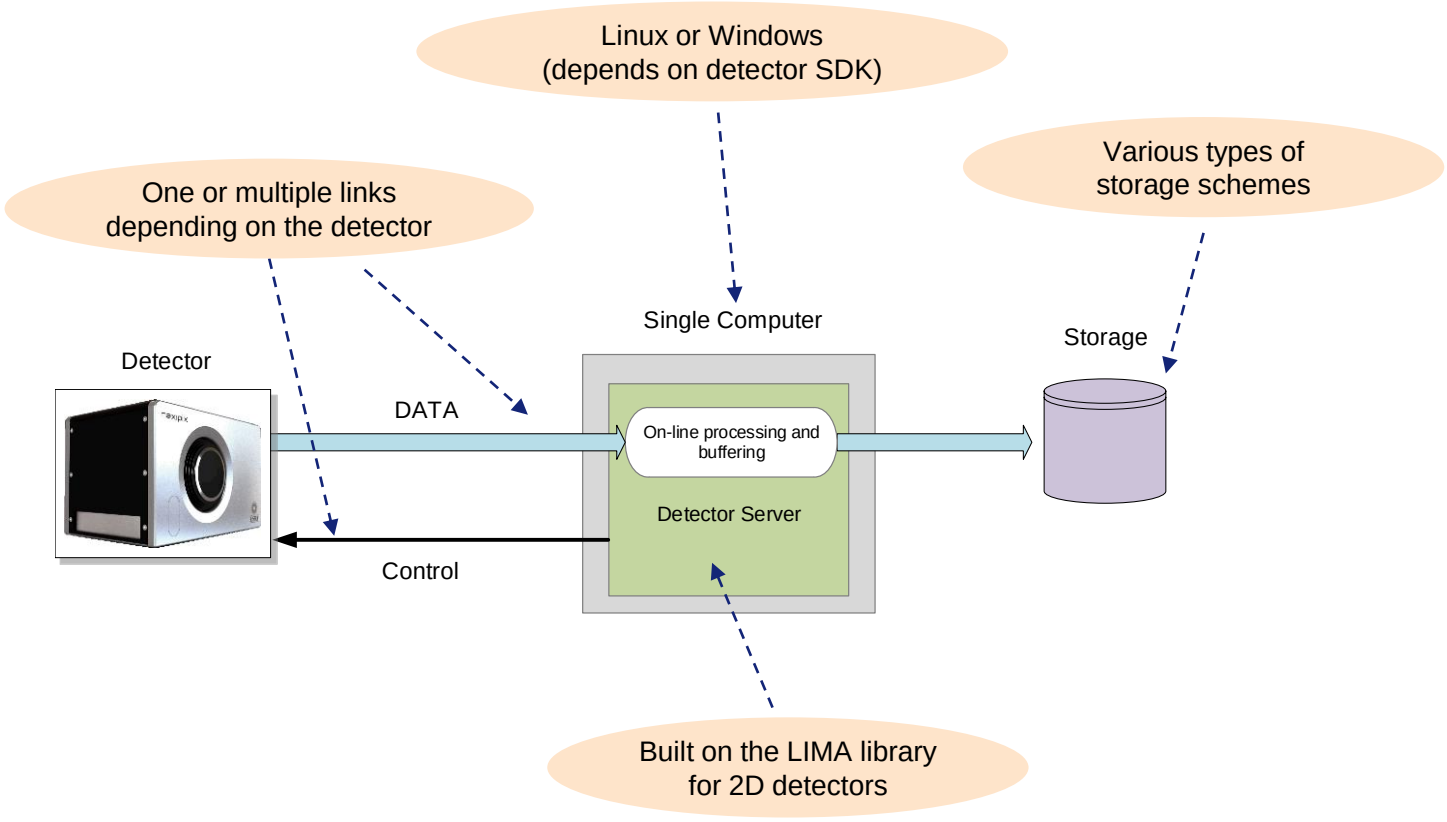
Non commercial



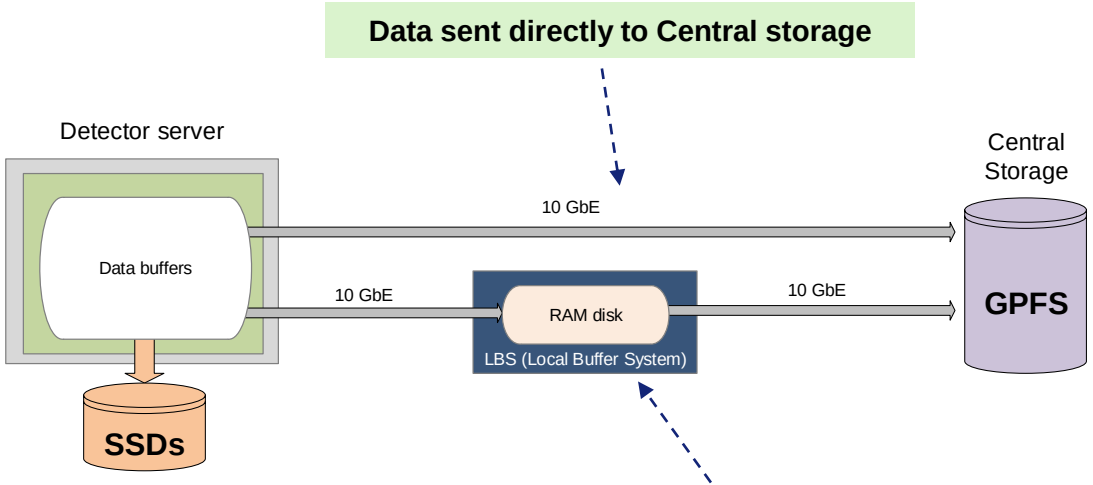
PSI/Eiger 2M

~ 8 GByte/s

BASIC DATA ACQUISITION SCHEME AT ESRF



VARIOUS DATA STORAGE SCHEMES



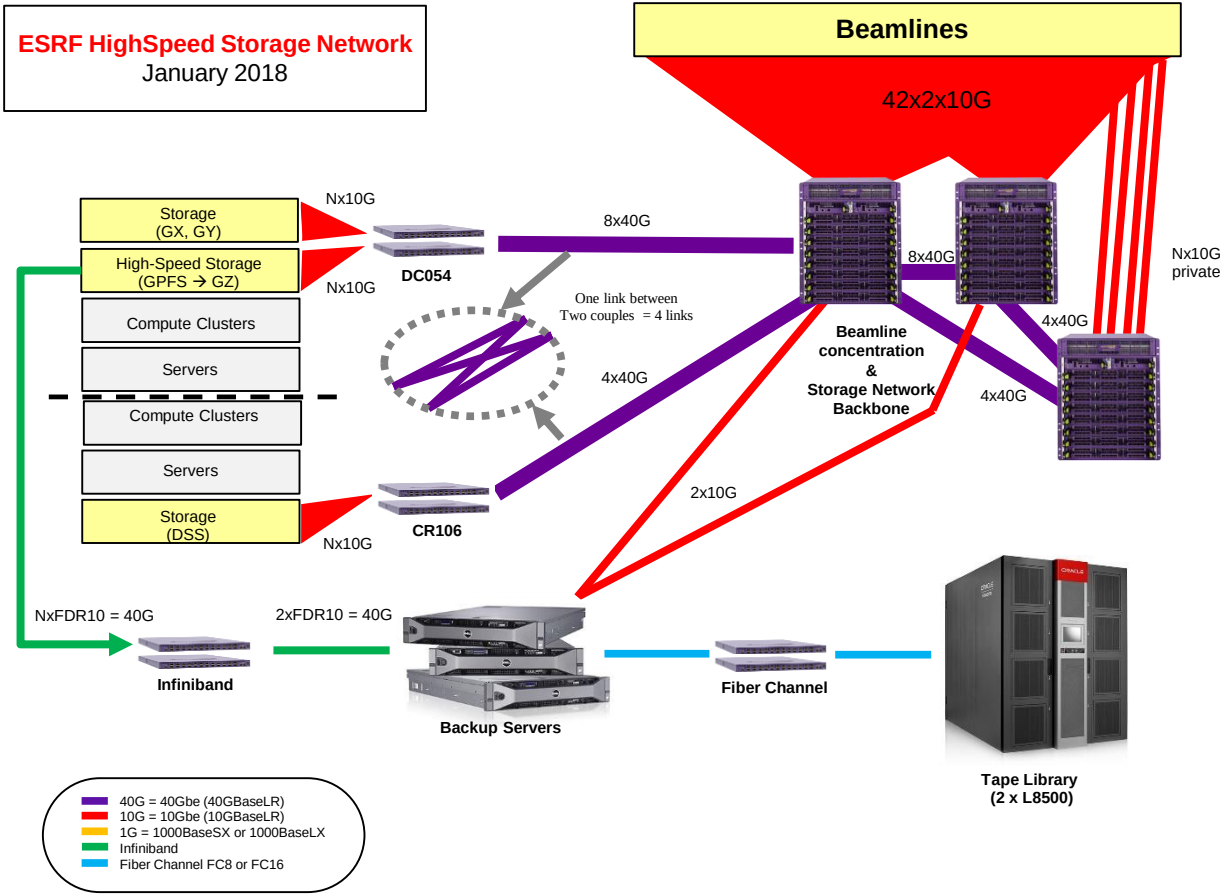
Data saved locally in the detector server

- Up to 1 TB SSD
- Transfer to central storage is initiated manually or semiautomatically

Local Buffer System (LBS)

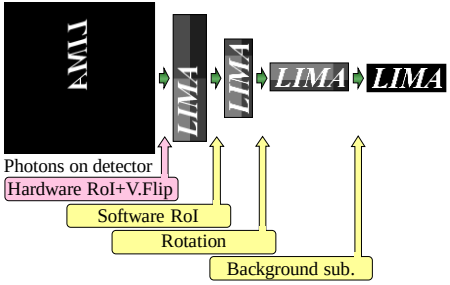
- Up to 540 GB RAM disks
- Transfer to central storage is automated

ESRF STORAGE NETWORK



DATA ANALYSIS

- Any low latency **on-line data treatment** is applied or managed by the detector server (via LIMA):
  - The LIMA library can implement a pipeline of data manipulation operations by itself that can be extended with plug-ins.



  - And in principle LIMA can 'delegate' to other processes for more complex or resource demanding processing (i.e GPU based). A few algorithms have been adapted to be included in the 'on-line processing pipeline' although they are still not in operation:
    - Azimuthal integration (powder diffraction , SAXS, …)
    - Time autocorrelation (XPCS)

- Today, in practice the data analysis processes and sequences take **data from disks** (storage)
  - In some cases (Tomography, MX, BioSAXS) data analysis is triggered by **automated workflows**
  - Although in most cases the analysis is **initiated manually** by the users.

- In more and more cases the users **cannot take the data with them** and do the analysis at home

DATA MANAGEMENT

Experimental data need to be properly managed to allow:

- Linking to publications
- Re-analysis
- Verification and anti-fraud
- New research
- Preservation of unique data sets
- Comply with EU Open Data requirements

Adoption and implementation of an official ‘Data Policy’

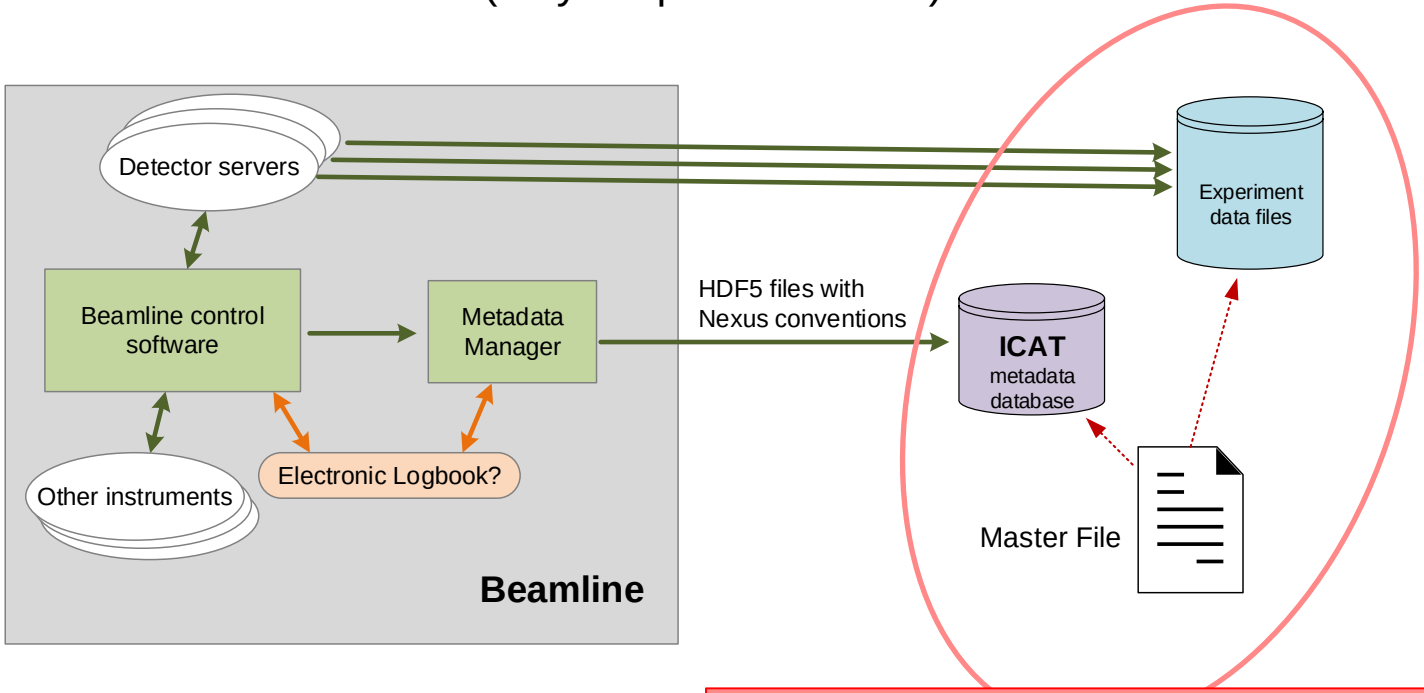
ESRF DATA POLICY (SUMMARY)

http://www.esrf.eu/datapolicy

1. The ESRF shall act as a **custodian** of the data
2. All raw data will be curated in a **well defined format**
3. **Metadata** is **captured automatically** and resides within the raw data files and or on-line catalogue
4. Access to raw data is **restricted** to the experimental team for a **maximum of 3 years** (embargo period)
5. Embargo period can be **extended** on request
6. ICAT will link the data to the proposal and publication
7. **Ownership** of all results (intellectual property) derived from the analysis of the raw data is determined by the contractual obligations of the person(s) performing the analysis
8. Analysis of openly accessible data **must acknowledge** the source of the data and cite its unique identifier and any publication linked to the same raw data.

METADATA COLLECTION AND BUILDING A SESSION DATA SET

(very simplified scheme)

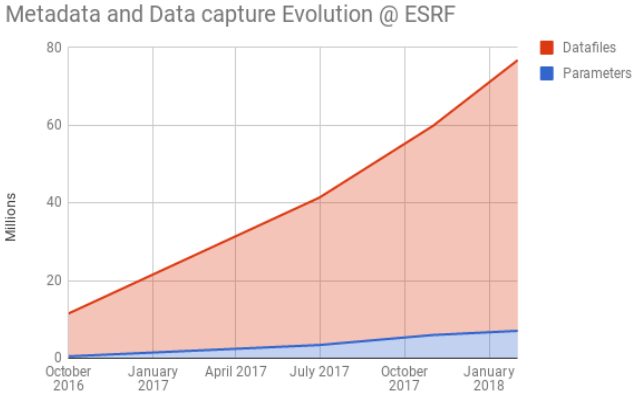


Experimental session data set

- Exposed as a single HDF5 file
- Assignment of a DOI (Digital Object Identifier)

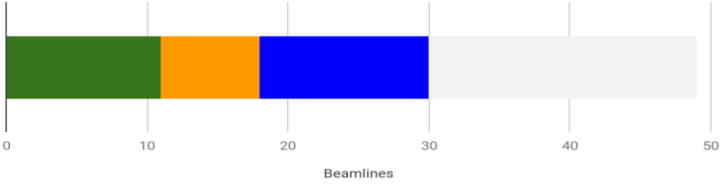
IMPLEMENTATION OF THE ESRF DATA POLICY

- **Metadata Collection**
  - Automatic capture of data and metadata
- **Data archiving**
  - Long term archiving in tape library **during 10 years**
- **Raw Data in HDF5**
  - HDF5 used as primary format for raw data
- **Open access of data**
  - **Persistent identifier** (DOI) associated to data from peer review proposals and open access data after an embargo period of 3 years



Current Status

- Data policy already implemented on 11 beamlines, 7 in progress and 12 planned for 2018



- **Ongoing developments for high-throughput data acquisition**
  - **Distributed LIMA library**
  - **RDMA based framework (RASHPA)**

LIMA: LIBRARY FOR IMAGE ACQUISITION

World-wide collaboration: synchrotrons, large facilities, R&D institutes, detector manufacturers. In ‘production’ since 2010

Among its Features:

- Provide **common user functionality**
- Separate hardware control from software tasks
- Data saving various file formats (EDS, HDF5, …)
- Includes a **multi-threaded processing framework**:

  Geometric transformations
  - ✓ Frame reconstruction, stripe concatenation
  - ✓ Rotation, Flipping, Binning, Region-of-Interest
  - ✓ Image masking

  Basic Image processing
  - ✓ Frame accumulation
  - ✓ Background subtraction, flat-field corrections

  Data compression (LZ4, gzip)

  User-defined operations can be added (plug-ins)
- Highly **optimised usage of computer resources**

Supported Detectors

- ESRF Frelon & Maxipix
- Dectris Pilatus2&3, Eiger
- GigE: Basler, PointGrey, Prosilica, Ueye
- Rayonix, ADSC, MarCCD
- STFC: Hexitec, Ultra, XH, Xspress3, Merlin
- PCO.dimax, edge, 2K, 4K
- Andor I-Kon, Zyla, Neo
- Hamamatsu Orca
- v4l2
- PerkinElmer, Dexela
- PSI detectors: Eiger 2M & 500K
- Lima Meta camera (4x Maxipix)
- Aviex, Pixirad, imXPAD

LIMITATIONS FOR HIGH DATA THROUGHPUT DAQ

Today at ESRF: data is streamed through a LIMA based 'detector server'

Processing very high throughput data streams is challenging because:

- **Single computer** (even though LIMA is highly multithreaded)
  - → **Multicomputer versions of detector servers** (distributed LIMA)
- All image manipulation is **100% software based**
  - → **Hardware assisted DAQ** and image manipulation (RASHPA)

Tuning the performance of the LIMA server to achieve full performance for a PSI/Eiger 500k module took **several months** to highly qualified DAQ software expert (Alejandro Homs)

LIMA Development Roadmap

General improvements:

- Better packaging and deployment
- Image display: flexible GUI layouts with SILX framework
- Data storage: Common API for different saving streams
- Introduce new data types (not only images)

High-performance detectors:

- Memory management: improved control of acquisition and processing buffers
- Include **branches in frame processing** pipeline
- **Multi-backend computer** support
  - Distributing full image frames among computers
  - Dispatching partial frames (modules?) to separate computers

RASHPA (HARDWARE ASSISTED DAQ)

RDMA-based Acquisition System for High Performance Applications

Project initiated within EU grants: **CRISP** (2011-2014), **EUCALL** (2015-2018)

- Development and validation of concept and demonstrators
- First implementation with a real detector is in progress (SMARTPIX)

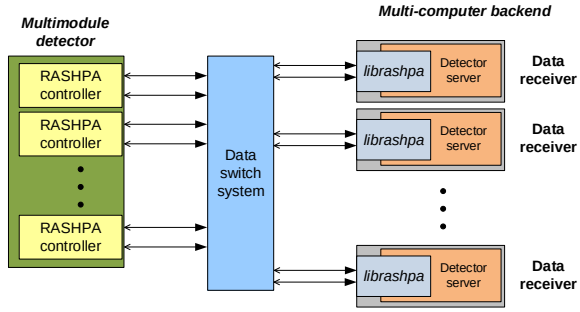
Key/special features:

- Data is pushed into destination by Remote DMA (Direct Memory Access)
  - **Zero-copy**, minimise software intervention
- Multiple data transfer processes can run **simultaneously**
  - Data dispatch various purposes: data storage, pre-processing, display, …
- Implements detector related **data manipulation** from the source (the detector)
  - Geometry related (image reconstruction/aggregation, ROI extraction, …)
- Software **configurable**
  - Number of data streams, destination buffers, data selection and dispatching

RASHPA COMPONENTS

Main components

- **RASHPA controller(s)** embedded in the detector
  - Implemented by CPU+FPGA
  - Each module must implement its controller
- **RDMA-capable data link**
  - High throughput and routable (switches)
- Backend computers (**System manager** + **Data receivers**)
  - Executing *librashpa* (Linux library)



Tested data links:

- **PCIe over cable** (copper or fiber optics)
  - Fits all the functional requirements
  - But too limited availability of commercial components

- **Ethernet**
  - Implementing efficient RDMA protocols is not so straightforward
    - RoCEv2 (UDP based) is the best candidate
  - But unbeatable in what respects to **availability and cost** of **high-performance** hardware
  - Recently validated our implementation of **100GbE UDP transfers** (FPGA-FPGA, FPGA-NIC)

THANK YOU!