RASHPA: a High Performance Data Transfer Framework for 2D Detectors

Pablo Fajardo
Fabien Le Mentec
Thierry Le Caer
Christian Herve
Alejandro Homs-Puron

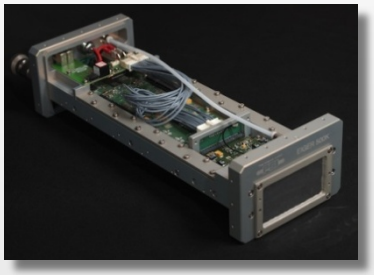
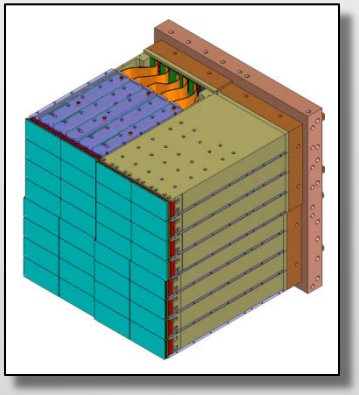
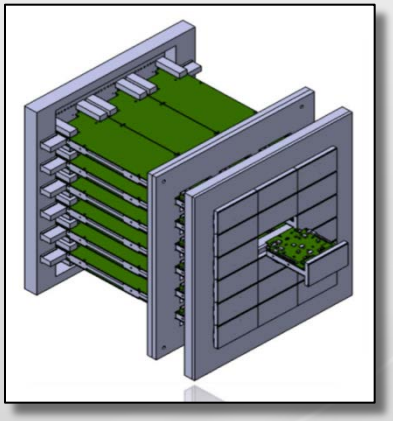
Detectors & Electronics / Software
Instrumentation Services and Development Division
ESRF

Talk outline

- **Introduction**
  - **Context**
  - **Motivation**
- **RASHPA Goals & Design**
  - **High performance**
  - **Functionality & Scalability**
- **Present & Near Future**
  - **Current status & Next steps**
  - **Conclusions**

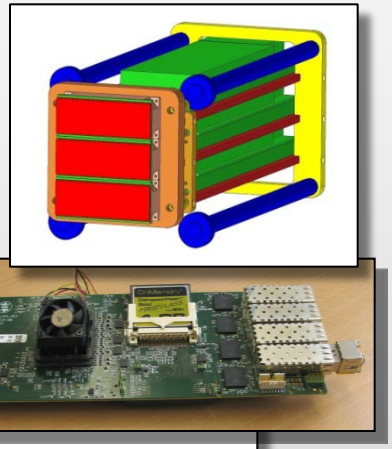
High performance detectors

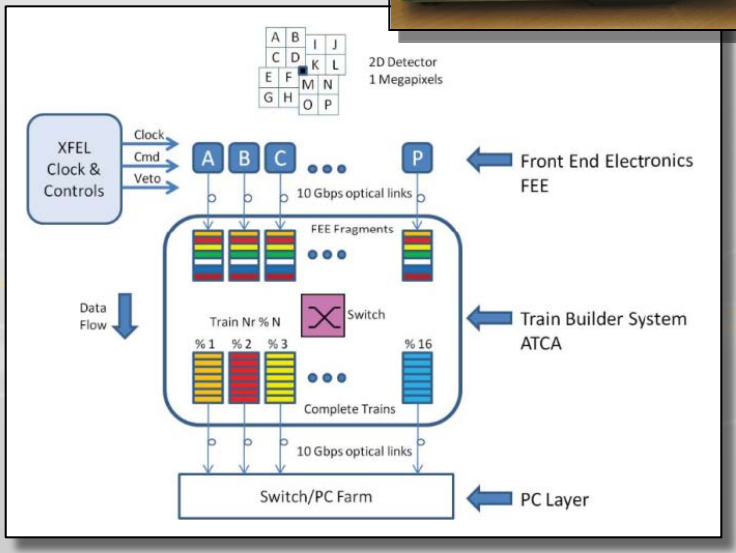
- Increase X-ray source brilliance
- High frame rate: 1 – 10 KHz
- Large detectors
- CMOS – Parallel ADCs: 10 GByte/s
- Pixel detectors
  - Single chips: 100 MByte/s
  - Basic modules: 0.5 – 1 GByte/s
  - Multi-module: 10+ GByte/s
- Data transfer can limit sensor speed
- Proprietary solutions


Data transfer solutions

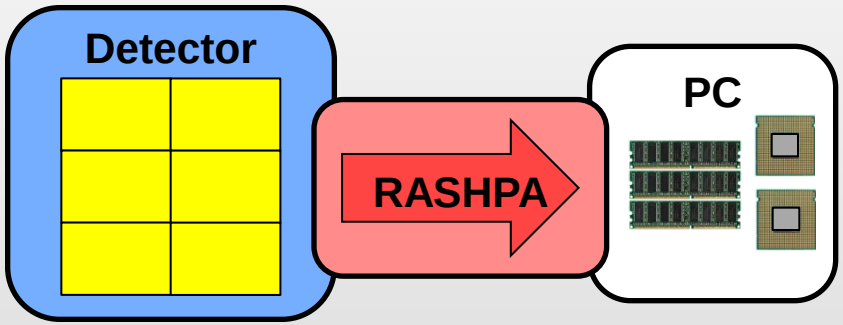
- Mostly based on 10 Gigabit Ethernet
- Specific protocols
- Detector-centric
- UDP transport simplifies FEE
- Complex frame builders
- Backend PC is not always a priority
- Difficult to optimize performance
  - Memory copy





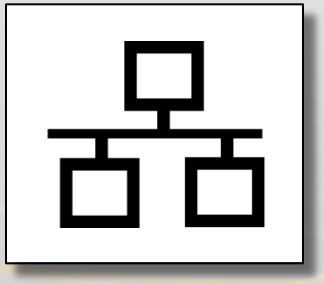
RASHPA Goals

- Generic data transfer framework
- Oriented to area detectors:
  - Asymmetric bandwidth
  - Knowledge of geometries
- Scalable:
  - Multi-link controllers
  - Multi-module detectors
- Parallel data streams:
  - Raw, Region-of-Interest, Live, Metadata
- Backend PC-friendly:
  - Efficient access to host memory
  - Minimal CPU overhead – event driven



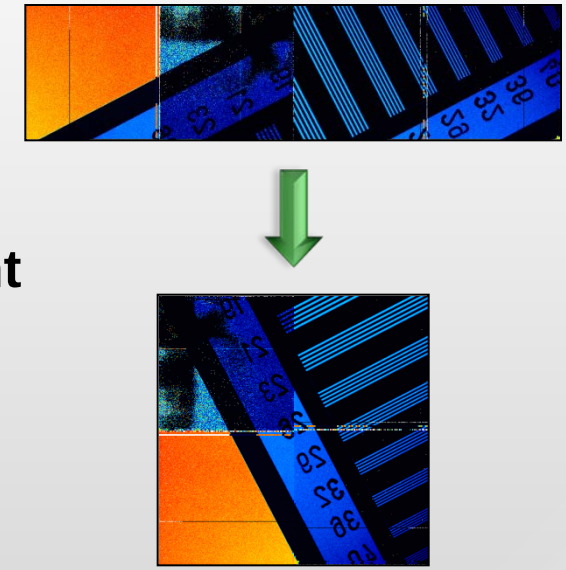
High speed technology

- Help detector readout design
- Simplify backend PC solutions
- Use industry standards
  - PCI-Express
  - 10/40 Gigabit Ethernet
  - … Infiniband
- Avoid proprietary protocols



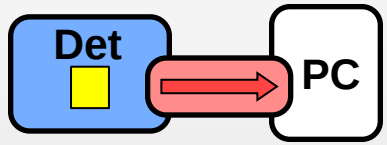
Detector Geometries

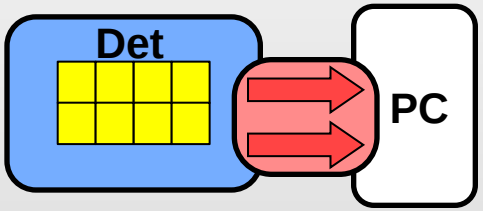
- Memory management consistency
- Chip & module tiling:
  - Detector frame reconstruction
  - Hardware writes in corresponding quadrant
  - Avoid unnecessary memory copies
- Image transformations are expensive to CPUs
  - Rotation, Flip, RoI
  - Much faster in hardware
- RASHPA can:
  - Perform some basic manipulations at transfer level
  - Coordinate advanced operations on the detector
  - Propagate geometries to memory management

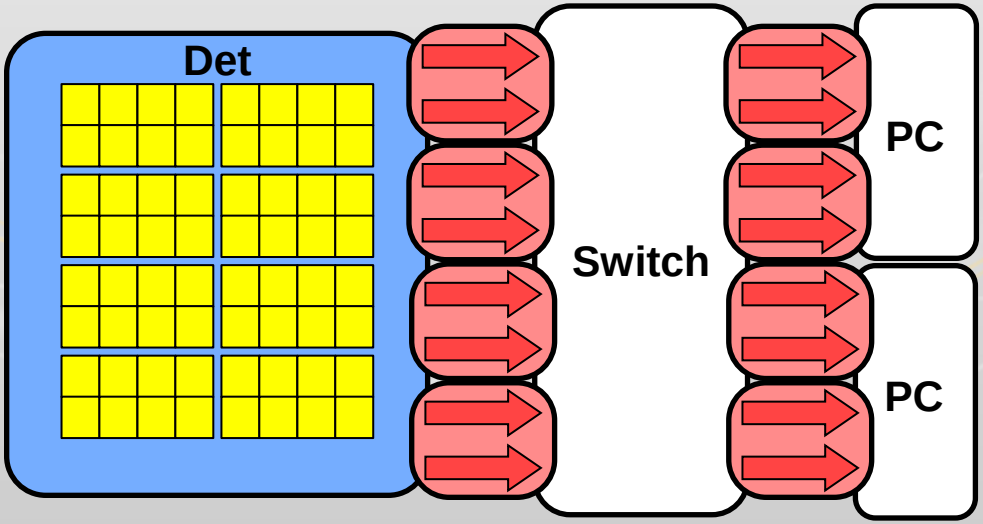


Scalable topology

- Multi-link controllers
- Low-profile detector ⇨ single link
- High-performance detector:
  - Full speed ⇨ need multiple links
  - Can work with fewer links ⇨ degraded mode
- Multi-module detector:
  - Multiple controllers
  - Connected to the same PC
  - Or to multiple PCs
  - Switches might be used



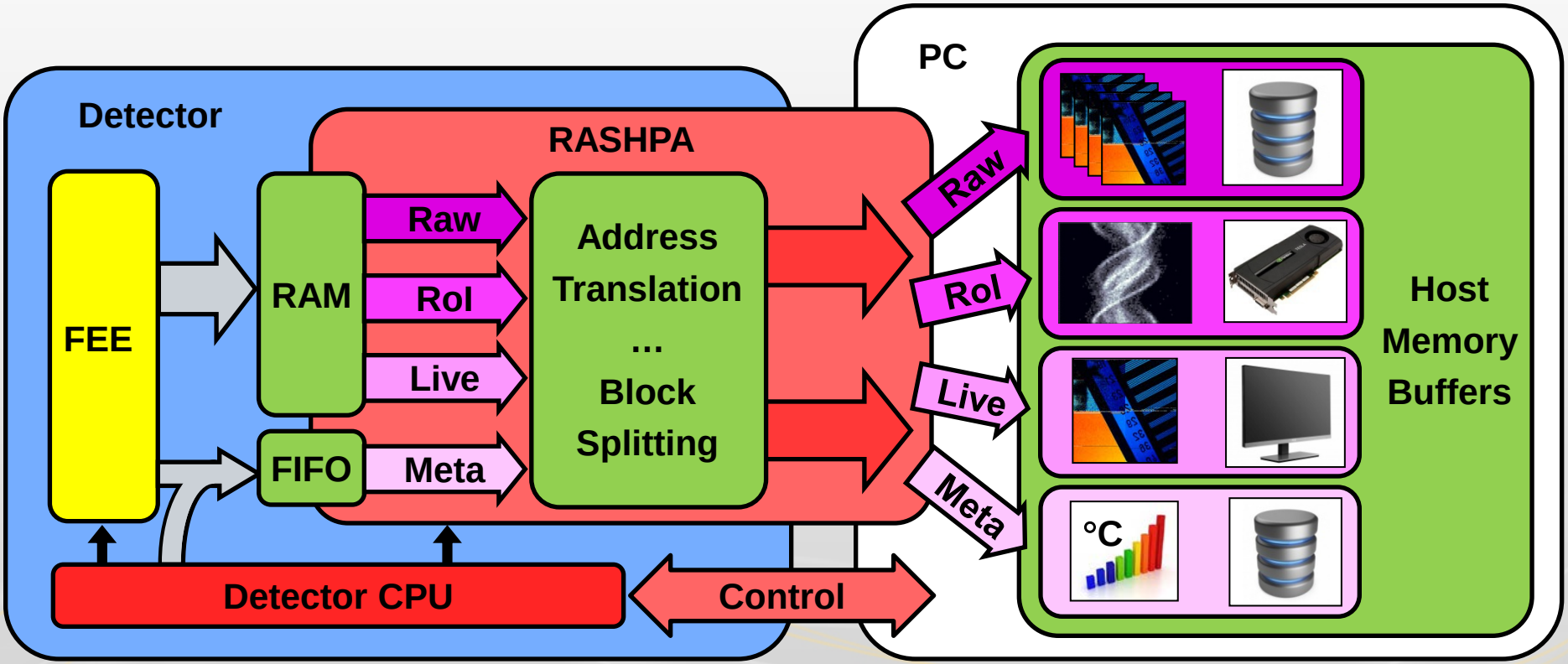




Robustness and Flexibility

- Data integrity
- Detector event notification
- Flow control
  - Strict overrun detection
- High level description meta-language
  - Detector capabilities & RASHPA configuration
- Multi-band detectors – data hierarchy
  - Variable block size – event list mode
- Different multi-buffer strategies
- Advanced transfer paths in the future
  - To Disks – NVM Express
  - To dedicated data processing board or GPU memory

Parallel Streams



Detector has a CPU controlling:

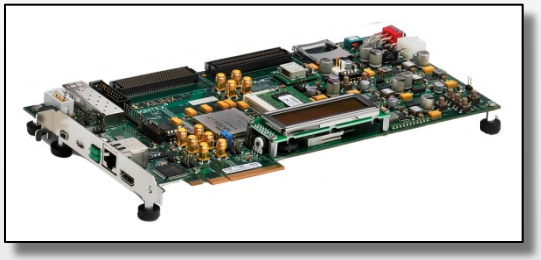
- Front End Electronics
- RASHPA

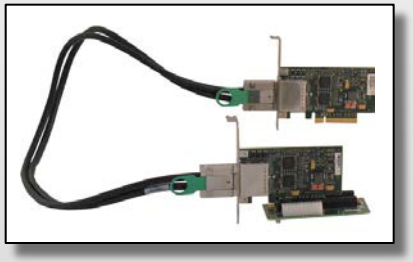
Multiple data streams:

- Dedicated buffer sets
- Different priorities

Current status

- Single-link demonstrator:
  - Xilinx Kintex 7 FPGA (KC705)
  - One Stop Systems Expansion Kit
  - PCI-Express Gen-2 x4
  - Re-use in-house FPGA architectures
- Under development:
  - FPGA firmware
  - Linux drivers & libraries
- 10 Gigabit Ethernet experience available
- Simulation environment using QEMU
  - Hardware described in VHDL or C





QEMU

Next steps

Final project deliverables:

- **Multi-link demonstrator, or**
- **Large-scale detector simulation**
  - **Validation of functionality**
  - **Not cycle-accurate, does not show true performance**
  - **Running exactly the same software**
- **RASHPA specification**

Conclusions

- RASHPA aims to be a high-performance, scalable architecture
- Not limited to fast data transport layers or protocols
  - Handle detector geometries, data hierarchy and dynamics
    - Help with image reconstruction
    - Provide parallel streams
  - Detector capabilities ⇨ Software auto-configure
  - Generic backend hardware and software solutions
- Single-link demonstrator is under development
- Future deliverables:
  - RASHPA specification
  - Multi-link demonstrator / simulation

Thank you for your attention!