The ALICE O^2 common driver for the C-RORC and CRU read-out cards

P. Boeschoten and F. Costa for the ALICE collaboration

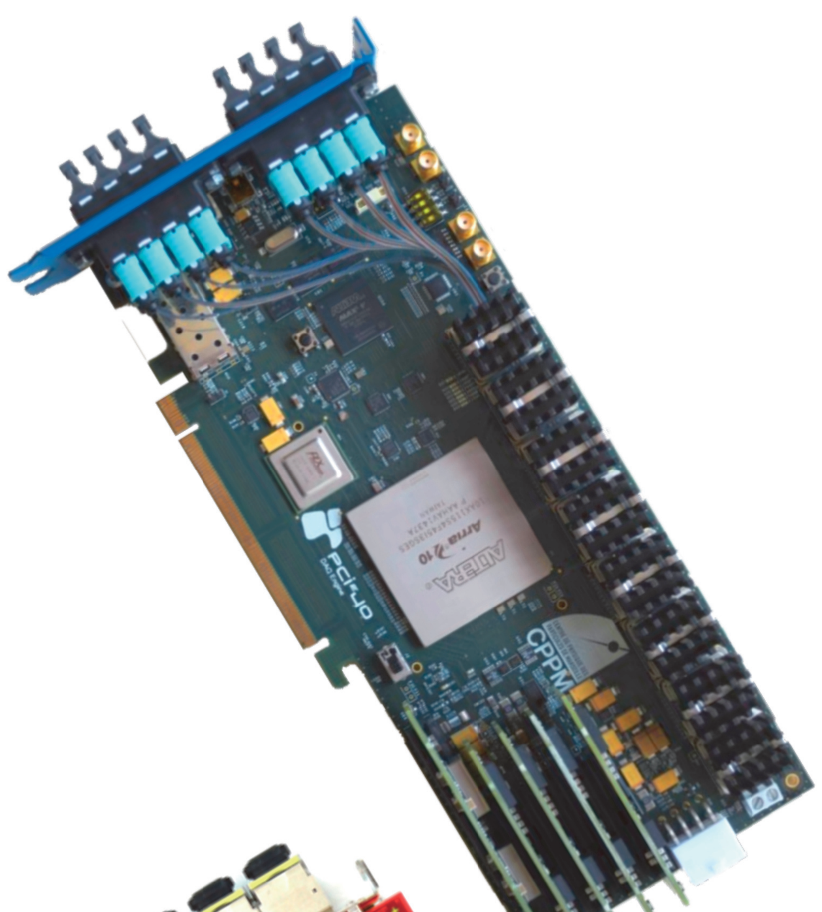
ALICE is the heavy-ion detector designed to study the strongly interacting state of matter realized in relativistic heavy-ion collisions at the CERN Large Hadron Collider (LHC). A major upgrade of the experiment is planned during the 2019-2020 long shutdown. In order to cope with a data rate 100 times higher than during LHC Run 2 and with the introduction of a continuous read-out mode, it is necessary to upgrade the Online and Offline Computing to a new common system called O^2. It will use commodity x86 Linux servers equipped with custom PCIe FPGA-based read-out cards.

Read-out cards

CRU

Common Read-out Unit

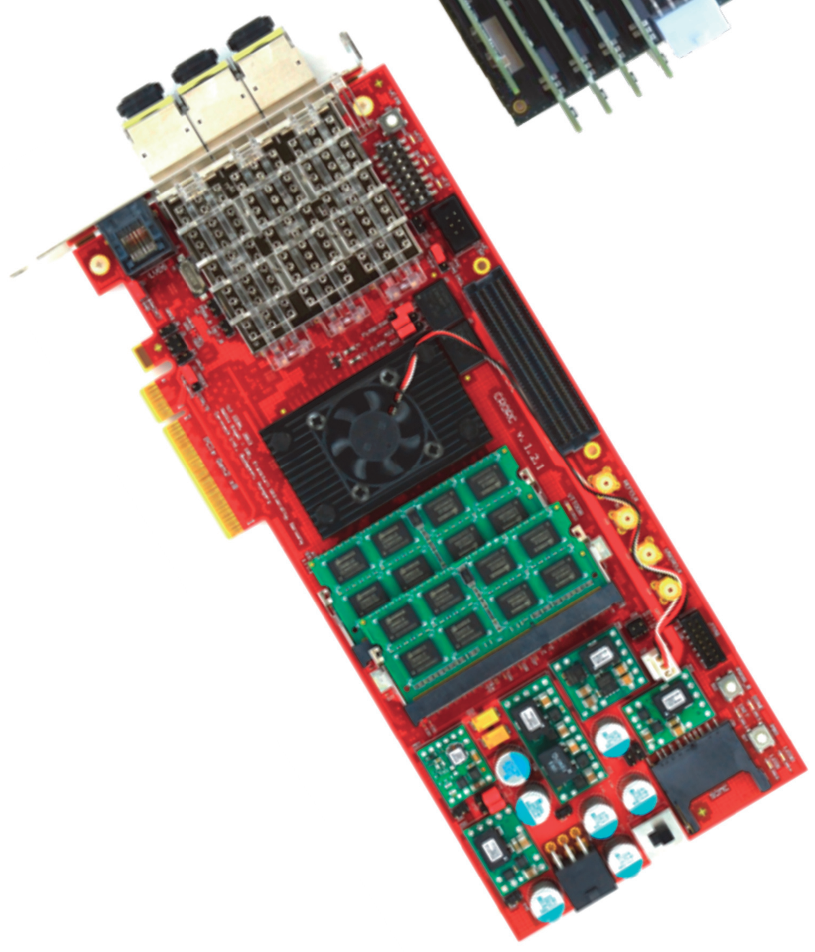
- PCIe v3x16
- 2 DMA channels
- 48 optical links
- ~27 GB/s throughput
  (may be further optimized)
- 461 units in ALICE for Run 3



C-RORC

Common Read-Out & Receiver Card

- PCIe v2x8
  (firmware implements v2x4)
- 6 DMA channels
- 12 optical links
- 1.6 GB/s throughput
- 25 units in ALICE for Run 3
- Reused from ALICE Run 2



Buffer structure

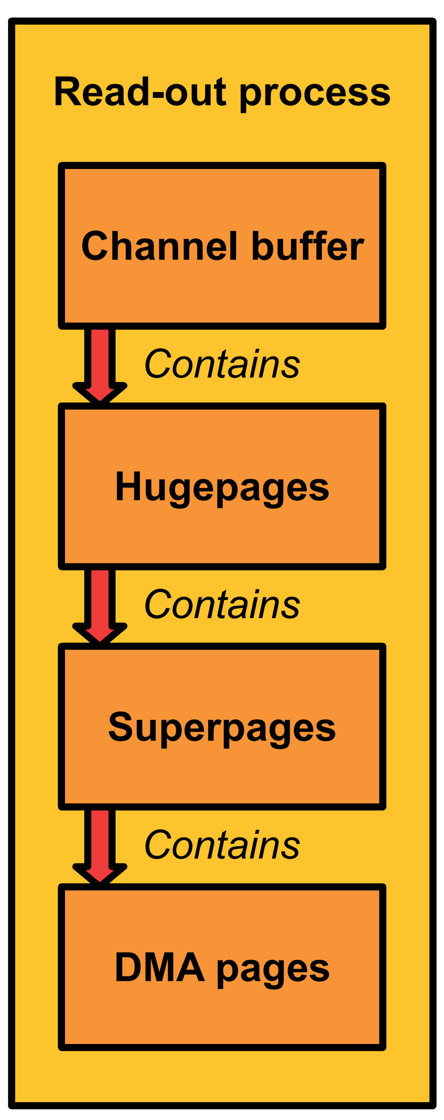
The buffer for the data is structured in a hierarchical manner for practical and performance reasons. It must be large to accommodate latency in the read-out chain, and ensure enough physical contiguousness to allow the DMA engine to work efficiently.

The Channel buffer contains all of the DMA data from one channel. It is allocated by the read-out task before starting a run
• *Typically many GiBs*

The Channel buffer is allocated using hugepages to ensure large, physically contiguous memory regions
• *2 MiB or 1 GiB*

The buffer is then divided into superpages, the logical units of transfer from the card
• *32 KiB to 1 GiB*

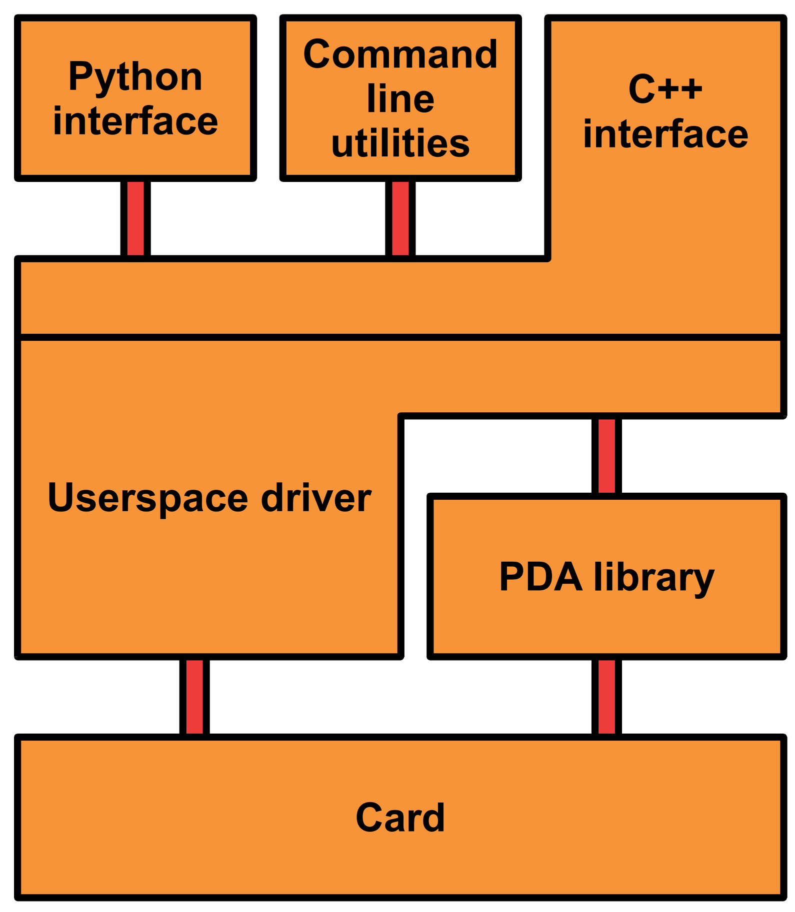
The superpages are filled by the card using DMA pages, the low-level unit of transfer of the card's DMA engine
• *8 KiB*



Software stack

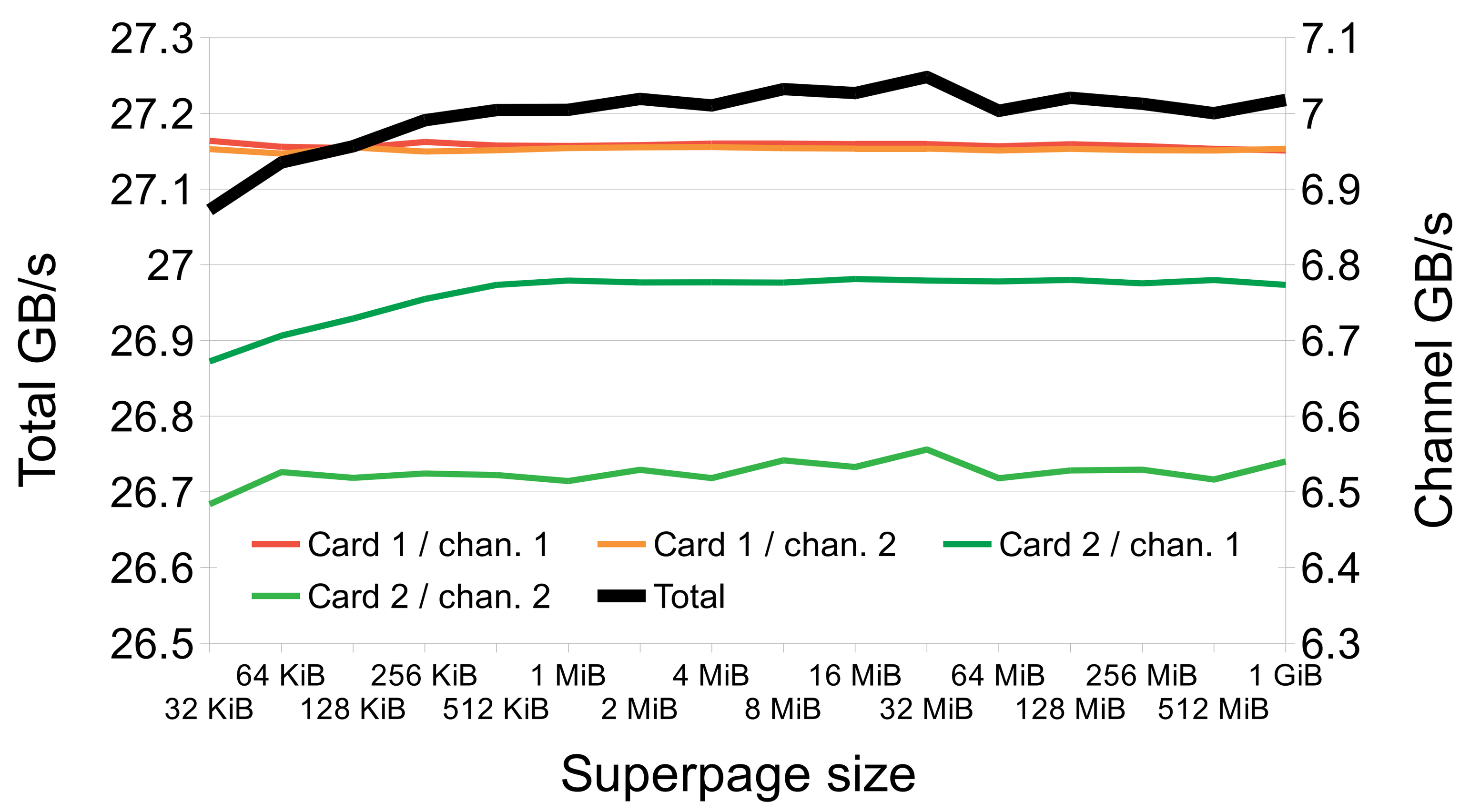
The software is subdivided into three layers:

- Low-level PCIe handling, provided by the PDA[1] library
- A userspace driver, and C++ interface to it for high-performance read-out
- A Python interface and command-line utilities that build upon the C++ interface



Performance

Throughput was measured on a Dell R730 server (2x Intel E5-2630 v3 CPUs, 2133 MHz DDR4 RAM) fitted with two CRU cards, using various superpage sizes, recording individual DMA channel throughputs and summing them to arrive at a total throughput for both cards. Little variance in total throughput is shown, even between the smallest supported superpage sizes and very large ones. More work is being done to understand the throughput differences between channels.



Reference & Contact

[1] **Portable Driver Architecture**
https://compeng.uni-frankfurt.de/index.php?id=173

[2] **ReadoutCard library**
https://github.com/AliceO2Group/FlpPrototype/tree/master/RORC

P. Boeschoten (author, presenter)
pascal.boeschoten@cern.ch

F. Costa (author)
filippo.costa@cern.ch