Computing at FAIR

Thorsten Kollegger

Facility for Antiproton and Ion Research & GSI Helmholtzzentrum
1 TByte/s into online farms

35 PByte/year on disk

300,000 cores at Tier 0

~100,000 cores distributed

Software Framework(s)
New European Accelerator Facility in Darmstadt, Germany

- 9 Member States + worldwide Partners
- Budget: 1 Billion Euros (2005)
- Upgraded accelerators from GSI, Darmstadt as injectors
Construction started in 2013
Full operation of the Modular Start Version (MSV) in the next decade
Synchrotrons: 1.1 km
HESR: 0.6 km
With beamlines: 3.2 km

Total area > 200 000 m²
Area buildings ~ 98 000 m²
Usable area ~ 135 000 m²
Volume of buildings ~ 1 049 000 m³
Substructure: ~ 1500 pillars, up to 65 m deep
FAIR Scientific Pillars

Four scientific pillars

• APPA: Atomic, Plasma Physics and Applications
• CBM: Compressed Baryonic Matter
• NUSTAR: Nuclear Structure, Astrophysics and Reactions
• PANDA: Antiproton Annihilations at Darmstadt
FAIR Experiments

APPA

CBM

PANDA

NUSTAR
CBM & PANDA
• ~500 members each
• HEP like detector systems

APPA & NUSTAR
• 700-800 members each
• Many small detectors / subcollaborations
A unifying element: Detector Readout

- Continuous readout with self-triggered front-end electronics
- Event definition & selection requires full reconstruction in online compute farms
  - No or limited hardware triggers
  - Convergence of on- and offline software
CBM Readout

- **on/near Detector**
  - FEE ASIC
    - e.g. STS-XYTER
  - Conc ASIC
    - GBTx
  - 320 Mbps

- **CBM E40**
  - FPGA
    - Preprocessing
      - Build micro slice containers
  - 4.8 Gbps

- **'Green Cube'**
  - FPGA
    - Buffering
      - Provide macro slice containers
    - CPU
  - 10 Gbps
  - 1 TByte/s

- Software
  - Slice Building
  - Reconstruction
  - Event Select
  - Monitoring
# PANDA time based simulation

<table>
<thead>
<tr>
<th></th>
<th>ev 1</th>
<th>ev 2</th>
<th>ev 3</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Wanted data structure

- **Slow detector**

## Data structure in Root

- **Event 1**
- **Event 2**
- **Event 3**

- **= detector hits from different events**

T. Kollegger - CHEP 2015
Online Computing Requirements

PANDA

300 GB/s
20M Evt/s

> 60 000 CPU-core or Equivalent GPU, FPGA ...

< 1 GB/s

CBM

1 TB/s

> 60 000 CPU-core or Equivalent GPU, FPGA ...

1 GB/s
ALICE LS2 Upgrade - Strategy

~1.1 TByte/s detector readout

*However:*

- storage bandwidth limited to ~20 GByte/s (design decision/cost)
- many physics probes have low S/B:
  - classical trigger/event filter approach not efficient

*Store only reconstruction results, discard raw data*

Data reduction by (partial) online reconstruction and compression

- >100,000 cores + GPUs + FPGAs

⇒ Implies much tighter coupling between online and offline reconstruction software
Some differences from FAIR to LHC/ALICE…

- No year-round data taking
- Experiments close (<1km) to FAIR Computing Center

Online Computing Clusters installed in FAIR Computing Center

- Tightly integrated and shared in FAIR Tier 0
Computing at FAIR

1 TByte/s into online farms
35 PByte/year on disk
300,000 cores at Tier 0
~100,000 cores distributed

Software Framework(s)
6 floors
Space for 768 19” racks (2.2m)
Max cooling power 12 MW
• 4 MW cooling (baseline)

Fully redundant (N+1)
No battery backup required

PUE <1.07

The FAIR Data Center (Green Cube)
Cooling Overhead
<7% of computer power

Works with every commercial computer

Primary Cooling circuit
makeupwater
cooling tower fixtures
cooling tower
equipment container

Secondary Cooling circuit
heat exchanger
primary pump
secondary pump

Overhead cooling
<7% of computer power works with every commercial computer.

Secondary cooling circuit: heat exchanger, primary pump, secondary pump, makeupwater, cooling tower fixtures, cooling tower.

Data center building: server room.
Construction started in Dec 2014
On schedule for completion in Q4/2015

Two weeks ago - full 6 floors completed this week
FAIR Tier 0

- Experiment Online Farms connected via ~4000 fibres to Detectors (~100,000 cores)
- Experiment Storage Systems (lustre based – 35 PByte/year)
- Tier 0 Processing Farm (~50,000 cores)
- Integration of HPC Centers of surrounding partners (Teralink) (opportunistic resource usage)
- Distribution of experimental data to FAIR Partners; several 100 Gb/s links
1 TByte/s into online farms

35 PByte/year on disk

300,000 cores at Tier0

~100,000 cores distributed

Software Framework(s)
New Challenges

Handling 1 TByte/s data transport in the online systems

Support increasingly parallel architectures
  • Multi-/Many-Core
  • SIMD

Support heterogeneous architectures
  • Accelerator cards (GPUs, Xeon Phi)

Picture from Herb Sutte: The Free Lunch Is Over A Fundamental Turn Toward Concurrency in Software Dr. Dobb’s Journal, 30(3), March 2005 (updated)
The L-CSC Cluster

Lattice-CSC:
• Built for Lattice-QCD simulations.

160 Compute nodes
• 4 * AMD FirePro S9150 GPU
• ASUS ESC4000 G2S Server
• 2 * Intel 10-core Ivy-Bridge CPU
• 256 GB DDR3-1600 1.35V
• FDR Infiniband
• 1.7 PFLOPS Peak
The L-CSC Cluster

Lattice-CSC (at GSI):

- Built for Lattice-QCD simulations.
- 160 Compute nodes
  - 4 * AMD FirePro S9150 GPU
  - ASUS ESC4000 G2S Server
  - 2 * Intel 10-core Ivy Bridge CPU
  - 256 GB DDR3-1600 1.35V
  - FDR Infiniband
- 1.7 PFLOPS Peak
- 5.27 GFLOPS/W
- First Rank in Green500 list in November 2014

T. Kollegger - CHEP 2015
Common simulation and reconstruction software framework for the FAIR experiments (and beyond)

https://github.com/FairRootGroup/FairRoot
Two projects – same requirements
Massive data volume reduction (1 TByte/s input)
• Data reduction by (partial) online reconstruction
• Online reconstruction and event selection
Much tighter coupling between online and offline reconstruction software
Common ALICE-FAIR Software Framework

A data-flow based model (Message Queues based multi-processing)

- Transport layer (FairMQ, based on: ZeroMQ, nanomsg)
- Configuration tools
- Management and monitoring tools
- Unified access to configuration parameters and databases.
FairRoot

ALFA

AliRoot6 (O²) CbmRoot R3BRoot SofiaRoot MPDRoot
ShipRoot PandaRoot AsyEosRoot FopiRoot EICRoot

Libraries and Tools

Protocol Buffers BOOST ROOT ZeroMQ CMake Geant4 Geant4_VMC Genat3 VGM

FairMQ Fair DB MC Application Module Detector Magnetic Field Event Generator Runtime DB
FairMQ

Multi-process concept with message queues for data exchange

• Each "Task" is a separate process, which:
  ▪ Can be multithreaded, SIMDized, …etc.
  ▪ Can run on different hardware (CPU, GPU, XeonPhi, …etc.)
  ▪ Can be written in any supported language (Bindings for 30+ languages)

• Different topologies of tasks can be adapted to the problem itself, and the hardware capabilities.

Balance between maintainability/development cost incl. training and performance
FairMQ

Parameter File(s) Database

Input File(s) Online Data

Setup Definition

Parameter Manager

Task 1
C

Task 2
Fortran

Task 3
C++

Task 4

Task 5

Task 6

Computer 1
Linux

Computer 2
Windows

GPU on Computer 3

Output File

Task 5

Task 4

T. Kollegger - CHEP 2015
One of the key challenges of the FairMQ approach: Process Management for 10,000 to 100,000 devices

- Control
- Monitoring
- Configuring

Dynamic Deployment System

- Separate module in FairRoot / ALFA
- XML description of process topology

http://dds.gsi.de/
Connecting the FairMQ devices/tasks requires knowledge of connection parameters

DDS supports dynamic configuration with key-value propagation

<table>
<thead>
<tr>
<th>Devices (user tasks)</th>
<th>startup time*</th>
<th>propagated key-value properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>2721 (1360 FLP + 1360 EPN + 1 Sampler)</td>
<td>17 sec</td>
<td>~ 6x10^6</td>
</tr>
<tr>
<td>5441 (2720 FLP + 2720 EPN + 1 Sampler)</td>
<td>58 sec</td>
<td>~ 23x10^6</td>
</tr>
<tr>
<td>10081 (5040 FLP + 5040 EPN + 1 Sampler)</td>
<td>207 sec</td>
<td>~ 77x10^6</td>
</tr>
</tbody>
</table>

* startup time - the time which took DDS to distribute user tasks, to propagate all needed properties, plus the time took devices to bind/connect and to enter into RUN state.
Kalman Filter (KF) Track Fit Library

Kalman Filter Methods

Kalman Filter Tools:
- KF Track Fitter
- KF Track Smoother
- Deterministic Annealing Filter

Kalman Filter Approaches:
- Conventional DP KF
- Conventional SP KF
- Square-Root SP KF
- UD-Filter SP
- Gaussian Sum Filter

Track Propagation:
- Runge-Kutta
- Analytic Formula

Conventional KF DP vs. SP

Conventional KF RK4 vs. Analytical

Implementations

Vectorization (SIMD):
- Header Files
- Vc Vector Classes
- ArBB Array Building Blocks
- OpenCL

Parallelization (many-cores):
- OpenMP
- ITBB
- ArBB
- OpenCL

Precision:
- single precision SP
- double precision DP

Strong many-core scalability of the Kalman filter library


with I. Kulakov, H. Pabst* and M. Zyzak (*Intel)
1 TByte/s into online farms
35 PByte/year on disk
300,000 cores at Tier0
~100,000 cores distributed

Software Framework(s)
FAIR Presentations at CHEP 2015

- Mohammad Al-Turany, ALFA: The new ALICE-FAIR software framework (#422)
- Ludovico Bianchi, Online tracking with GPUs at PANDA (#363)
- Gianluigi Boca, The Pattern Recognition software for the PANDA experiment (#512)
- Ivan Kisel, 4-Dimensional Event Building in the First-Level Event Selection of the CBM Experiment (#538)
- Dmytro Kresan, Online/Offline reconstruction of trigger-less readout in the R3B experiment at FAIR (#425)
- Florian Uhlig, New developments in the FairRoot framework (#258)
- Aram Santogidis, Optimizing the transport layer of the ALFA framework for the Intel Xeon Phi co-processor (#27)
- Suman Sau, High Speed Fault Tolerant Secure Communication for Muon Chamber using FPGA based GBTx Emulator (#436)
- Vikas Singhal, Event Building Process for Time streamed data (#507)
- Tobias Stockmanns, Continuous Readout Simulation with FairRoot on the Example of the PANDA Experiment (#319)
- Matthias Richter, A design study for the upgraded ALICE O2 computing facility (#439)
- Alexey Rybalchenko, Efficient time frame building for online data reconstruction in ALICE experiment (#353)