ALFA: ALICE-FAIR new message queuing based framework

Mohammad Al-Turany
GSI scientific computing

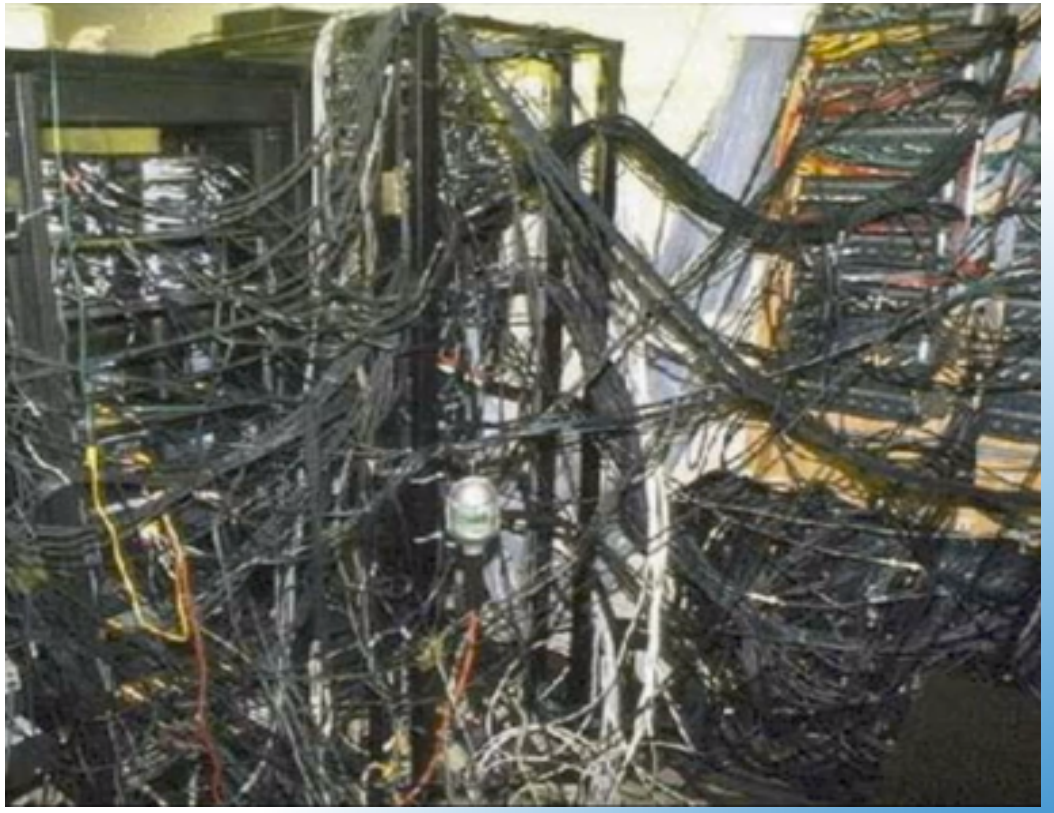
Developed in common by FairRoot Group (GSI), FAIR experiments and ALICE

What is new in ALFA compared to FairRoot?

ALFA has a data-flow based model: Message Queues based multi-processing

Works across all types of networks and more!

- Ethernet
  - ZMQ, nanomsg
- InfiniBand (IPoverIB, RDMA)
  - ZMQ, nanomsg, OFI
- Shared Memory Transport
  - Boost



Works across all types of networ
and more!

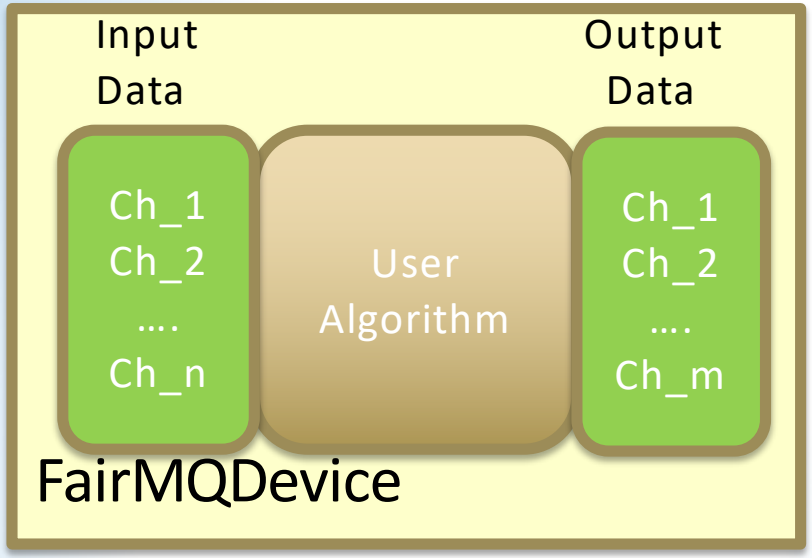
- Ethernet
  - ZMQ, na
- In
  - Shared Memory T
  - Boost
- RDMA)

RDMA-accelerated data transport in ALFA,
Dennis Klein (Poster 366)

Shared Memory Transport for ALFA,
Alexey Rybalchenko (Poster 305)

ALFA building block (FairMQ Devices)

- Message Queues for input/output
- Device takes/passes ownership of data
- Framework user sees only the callback to his algorithm
- Different channels can use different transport engines



Actor Model

Standalone processes ("devices") perform a task (e.g. track finding, fitting) and communicate with each other via messages (mediated by a queue).

Actor Model

- No locking, each process runs with full speed
- Easier to scale horizontally to meet computing and throughput demands (start/add new instances)

Right tool for the right job!

Each "Task" is a separate process, which:

- Can be multithreaded, SIMDized, …etc.
- Runs on different hardware (CPU, GPU, …, 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

Message format ?

The framework does not impose any format on messages.

It supports different serialization standards

- BOOST C++ serialization
- Google's protocol buffers
- ROOT
- Flatbuffers
- MessagePack
- User defined

Message format ?

The framework does not impose any format on messages.

It supports different so

- BOOST G
-
- essagePack
- User defined

Exploring polyglot software frameworks in ALICE with FairMQ and fer
Sebastien Binet (T5, Monday ,14:30 - 14:45)

How to deploy ALFA on a laptop, few PCs or a cluster?

DDS: Dynamic Deployment System

Users describe desired tasks and their dependencies using topology (graph) files

Users are provided with a WEB GUI to create topology (Can be created manually as well).

http://dds.gsi.de/

How to deploy ALFA on a laptop, few PCs or a cluster?

DDS: Dynamic [illegible]stem

[illegible]desired tasks and their [illegible]encies using topology (graph) files

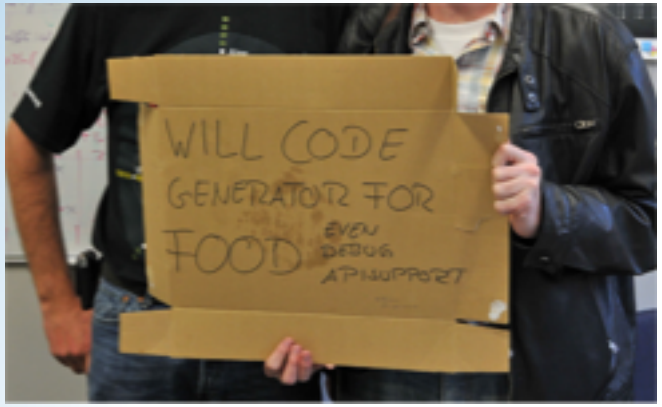
DDS – The Dynamic Deployment System
Andrey Lebedev (Poster 407)

Users are provided with a WEB GUI to create topology (Can be created manually as well).

http://dds.gsi.de/

ALFA in practice

ALFA can be easily used, adapted and extended even by typical HEP programmers.



ALICE Data Processing Layer (DPL):

Bring the ALFA Transport (FairMQ) with the ALICE O2 Data Model to provide ALICE specific, data flow oriented, API to the user.

ALFA can be easily used, adapted and extended even by typical HEP programmers.

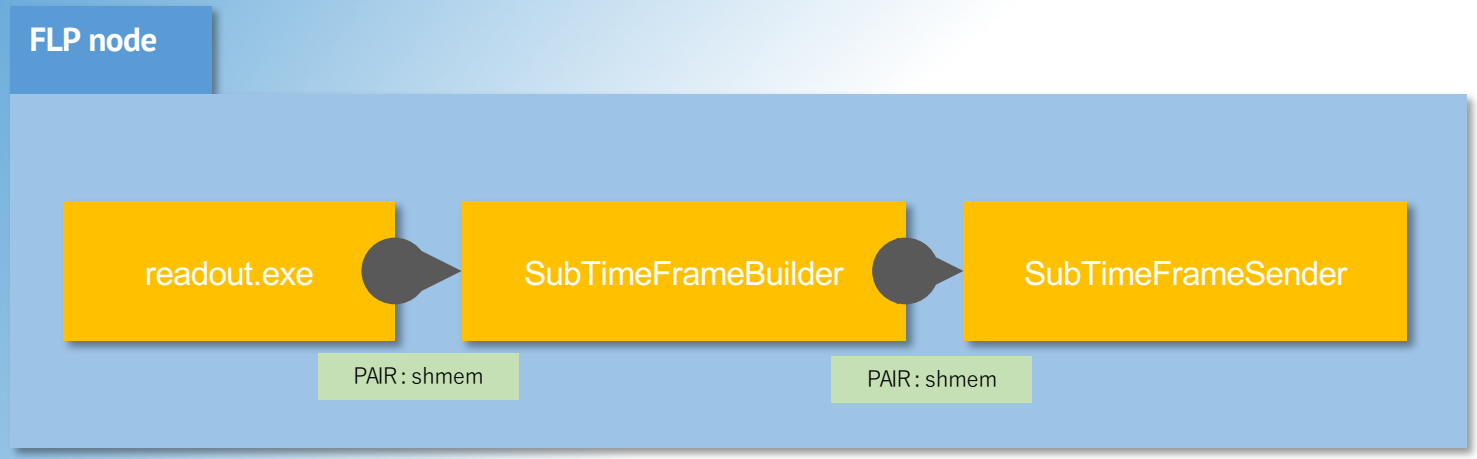
Evolution of the ALICE Software Framework for LHC Run 3
Giulio Eulisse (T5, Tuesday ,14:15 - 14:30)

ata Processing Layer (DPL):

Bring the ALFA Transport (FairMQ) with the ALICE O2 Data Model to provide ALICE specific, data flow oriented, API to the user.

Online Reconstruction

FLP collection (Readout)

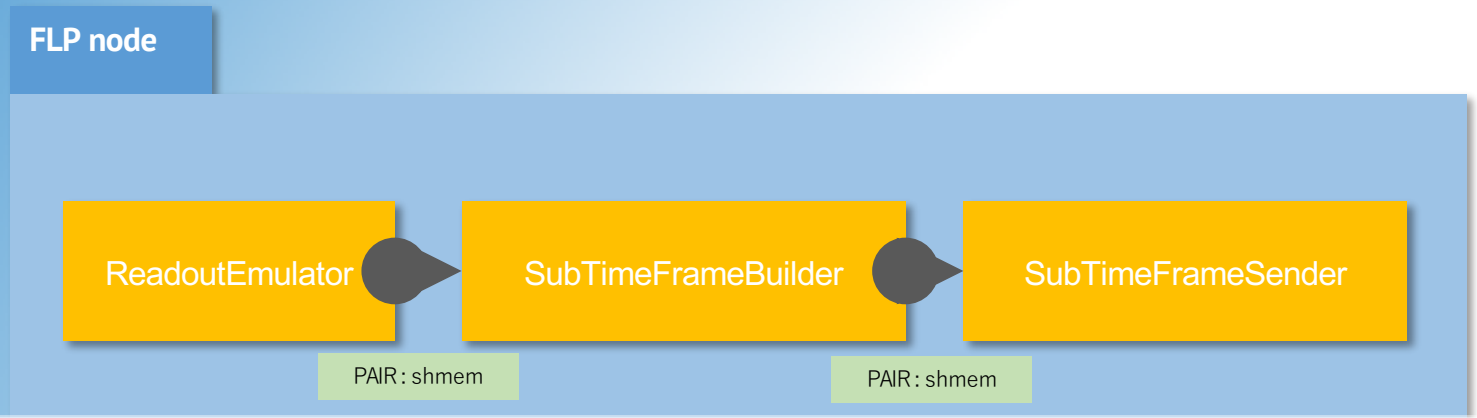



```
16      <decltask id="Readout"> <!-- Readout -->
17          <exe reachable="true">$INFOLOGGER_ROOT/bin/readout.exe file://$O2_ROOT/bin/config/readout_emu.cfg</exe>
18      </decltask>
```

```
20      <decltask id="SubTimeFrameBuilder"> <!-- SubTimeFrameBuilder to run with Readout -->
21          <exe reachable="true">$O2_ROOT/bin/SubTimeFrameBuilderDevice --id stf_builder_%taskIndex% --session ${session} --transport shmem --
22          <properties>
23              <id access="read">builder-stf-channel</id>
24          </properties>
25      </decltask>
```

```
42      <decltask id="SubTimeFrameSender">
43          <exe reachable="true">$O2_ROOT/bin/SubTimeFrameSenderDevice --id stf_sender_%taskIndex% --session ${session} --transport shmem --sh
44          <properties>
45              <id access="write">builder-stf-channel</id>
46              <id access="write">sender-stf-channel</id>
47          </properties>
48      </decltask>
```

FLP collection (Readout Emulator)

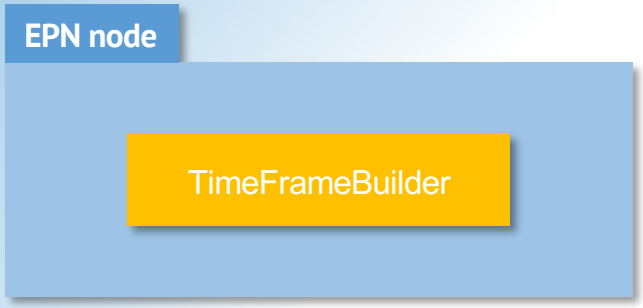



```
27      <decltask id="ReadoutEmulator"> <!-- Readout EMULATOR -->
28          <exe reachable="true">$O2_ROOT/bin/ReadoutEmulatorDevice --id readout-%taskIndex% --transport shmem --shm-monitor true --color 0 --
29          <properties>
30              <id access="write">readout</id>
31          </properties>
32      </decltask>
```

```
34      <decltask id="SubTimeFrameBuilderEmu"> <!-- SubTimeFrameBuilder to run with Readout EMULATOR -->
35          <exe reachable="true">$O2_ROOT/bin/SubTimeFrameBuilderDevice --id stf_builder_%taskIndex% --session ${session} --cru-count 1 --tran
36          <properties>
37              <id access="read">readout</id>
38              <id access="read">builder-stf-channel</id>
39          </properties>
40      </decltask>
```

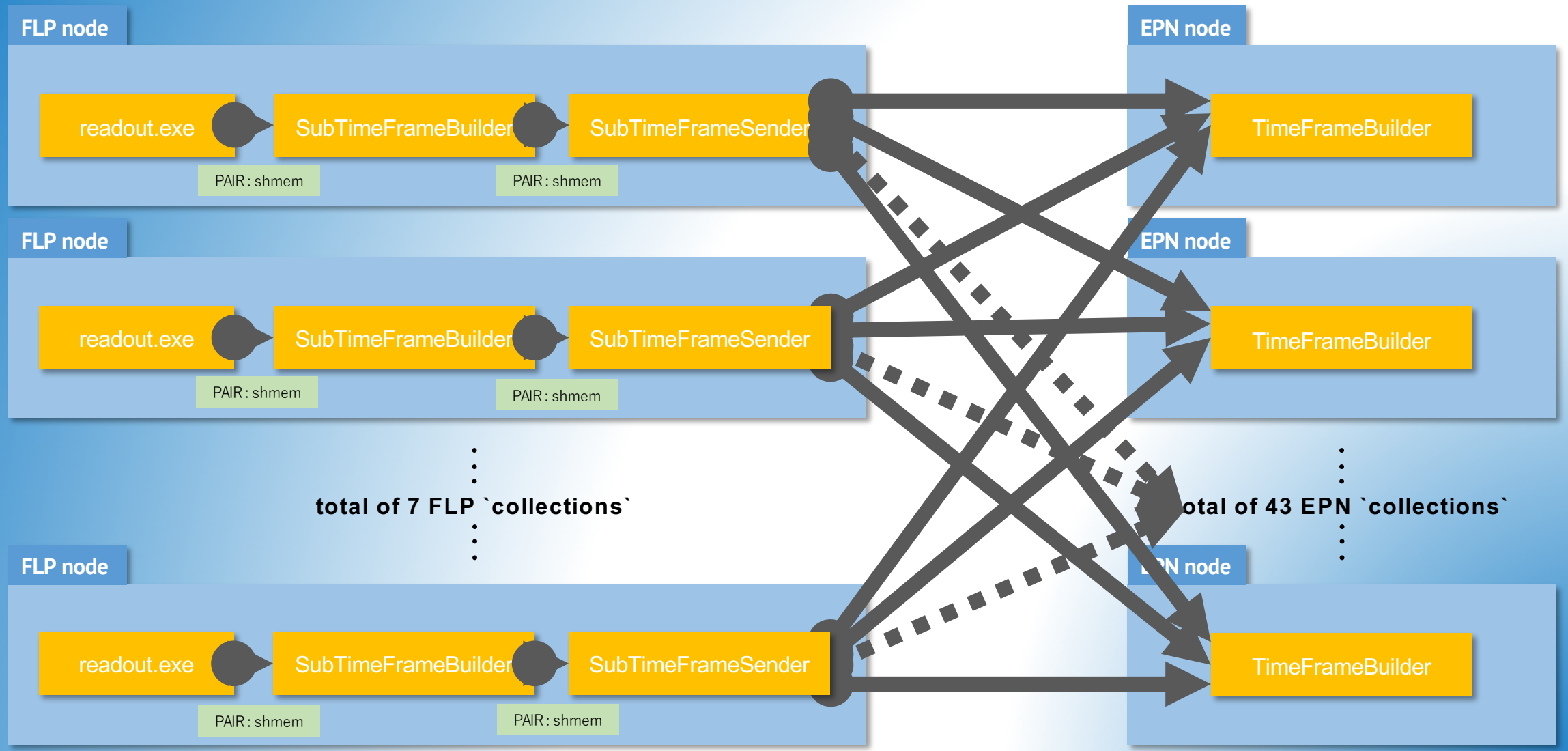
```
42      <decltask id="SubTimeFrameSender">
43          <exe reachable="true">$O2_ROOT/bin/SubTimeFrameSenderDevice --id stf_sender_%taskIndex% --session ${session} --transport shmem --sh
44          <properties>
45              <id access="write">builder-stf-channel</id>
46              <id access="write">sender-stf-channel</id>
47          </properties>
48      </decltask>
```

EPN Collection




```
50      <decltask id="TimeFrameBuilder">
51          <exe reachable="true">$O2_ROOT/bin/TimeFrameBuilderDevice --id tfbuilder_%taskIndex% --session ${session} --shm-monitor true --flp-
52          <requirements>
53              <id>epnhosts</id>
54          </requirements>
55          <properties>
56              <id access="read">sender-stf-channel</id>
57          </properties>
58      </decltask>
```

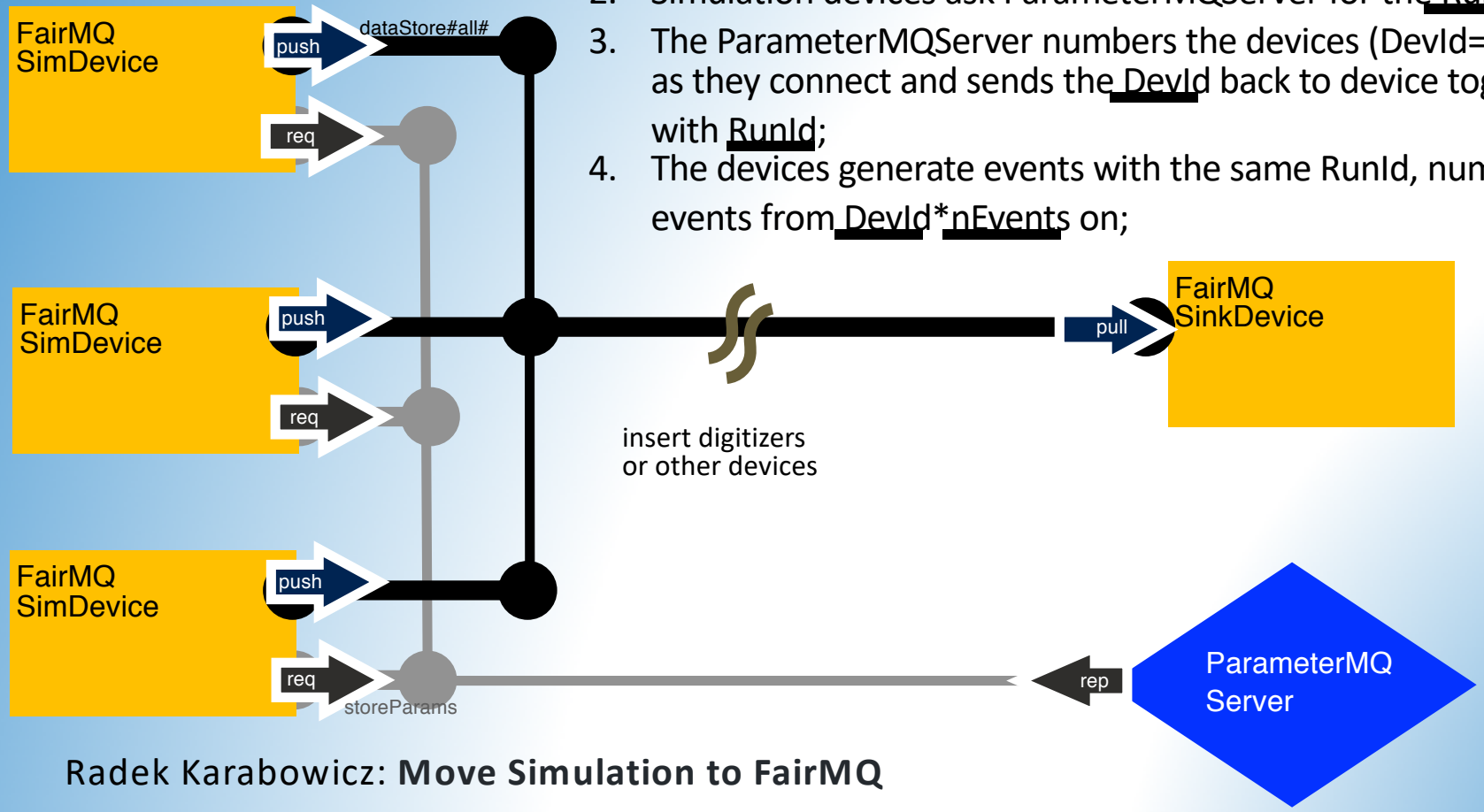
The O2 Prototype



ALFA/FairMQ in Simulation

Simulation in MQ

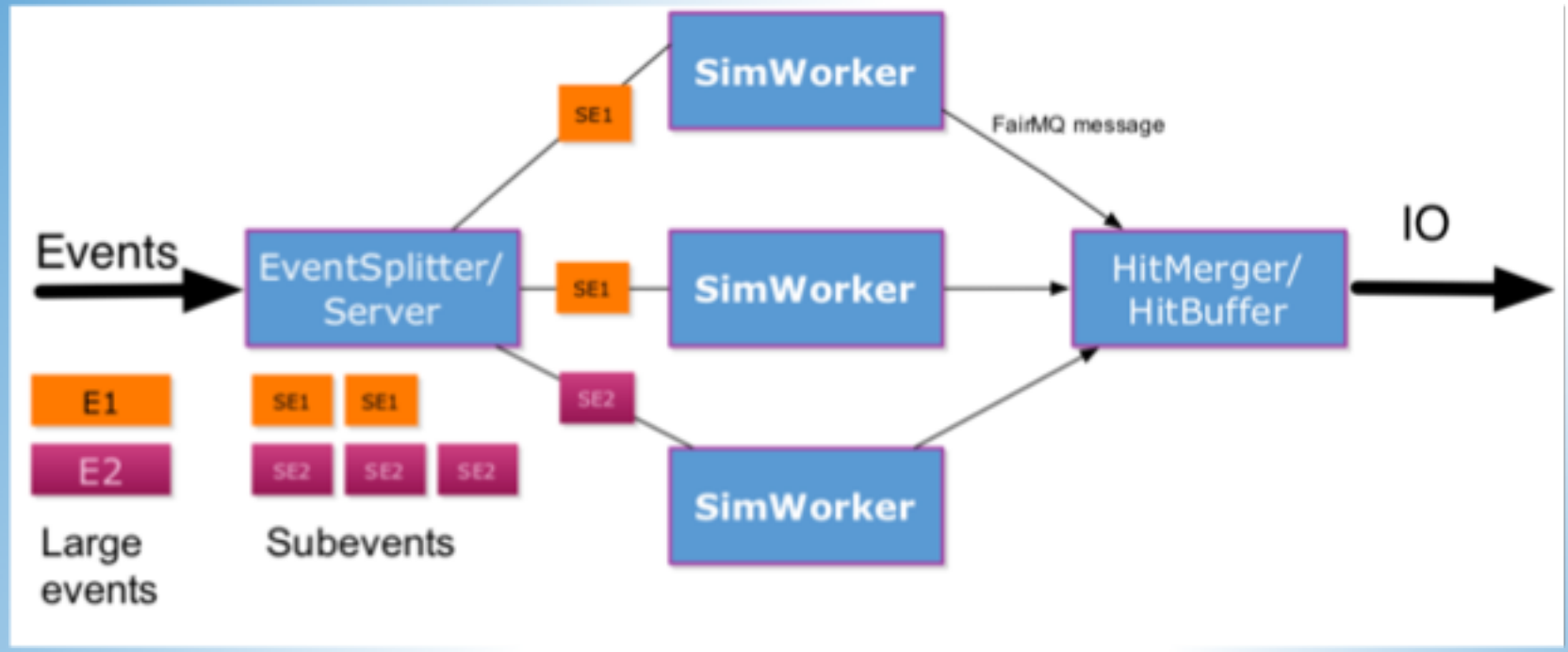
1. Each simulation device will generate nEvents;
2. Simulation devices ask ParameterMQServer for the RunId;
3. The ParameterMQServer numbers the devices (DevId=0,1,2,…) as they connect and sends the DevId back to device together with RunId;
4. The devices generate events with the same RunId, numbering events from DevId*nEvents on;



Radek Karabowicz: **Move Simulation to FairMQ**

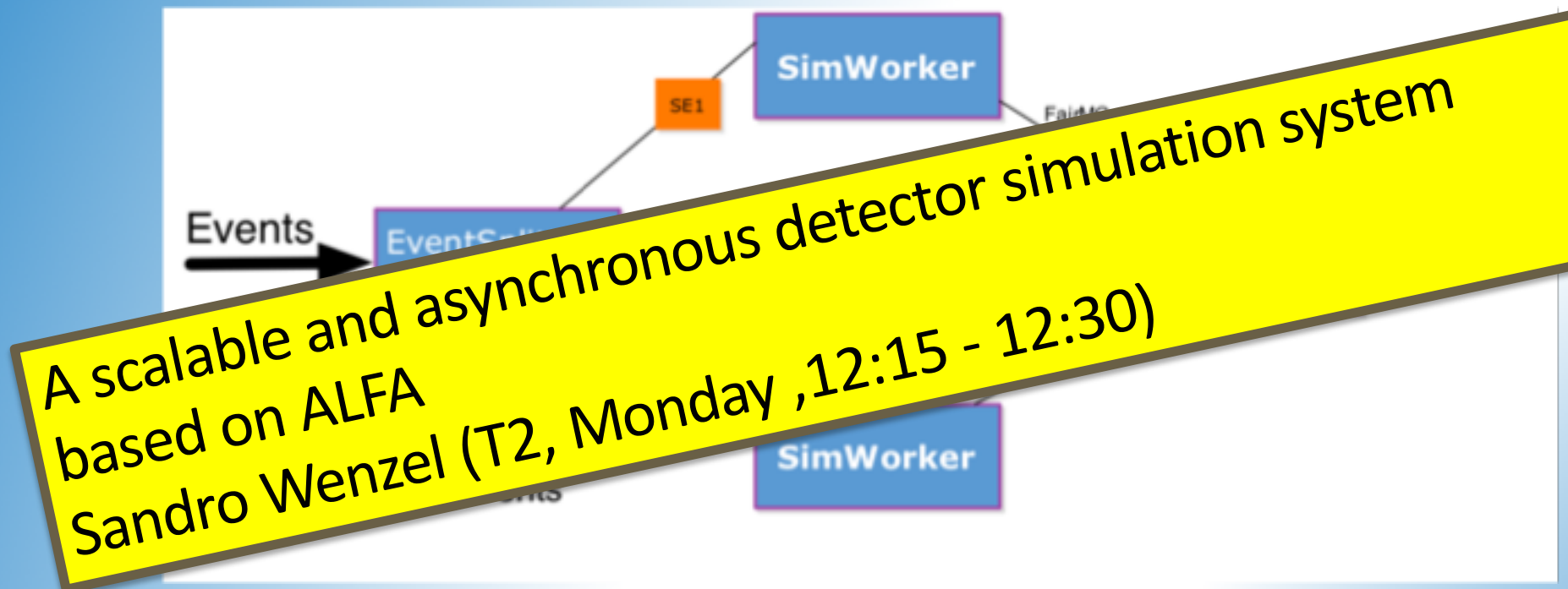
https://github.com/FairRootGroup/FairRoot/tree/dev/examples/MQ/pixelDetector

FairMQ-based parallel simulation



Sandro Wenzel

FairMQ-based parallel simulation



A scalable and asynchronous detector simulation system based on ALFA
Sandro Wenzel (T2, Monday ,12:15 - 12:30)

Summary

- ALFA allows developers to write their specific code in whatever language they choose as long as that language can send and receive data through message queues.

- ALFA provides an environment to write message passing processes, abstracting away many of the implementation details

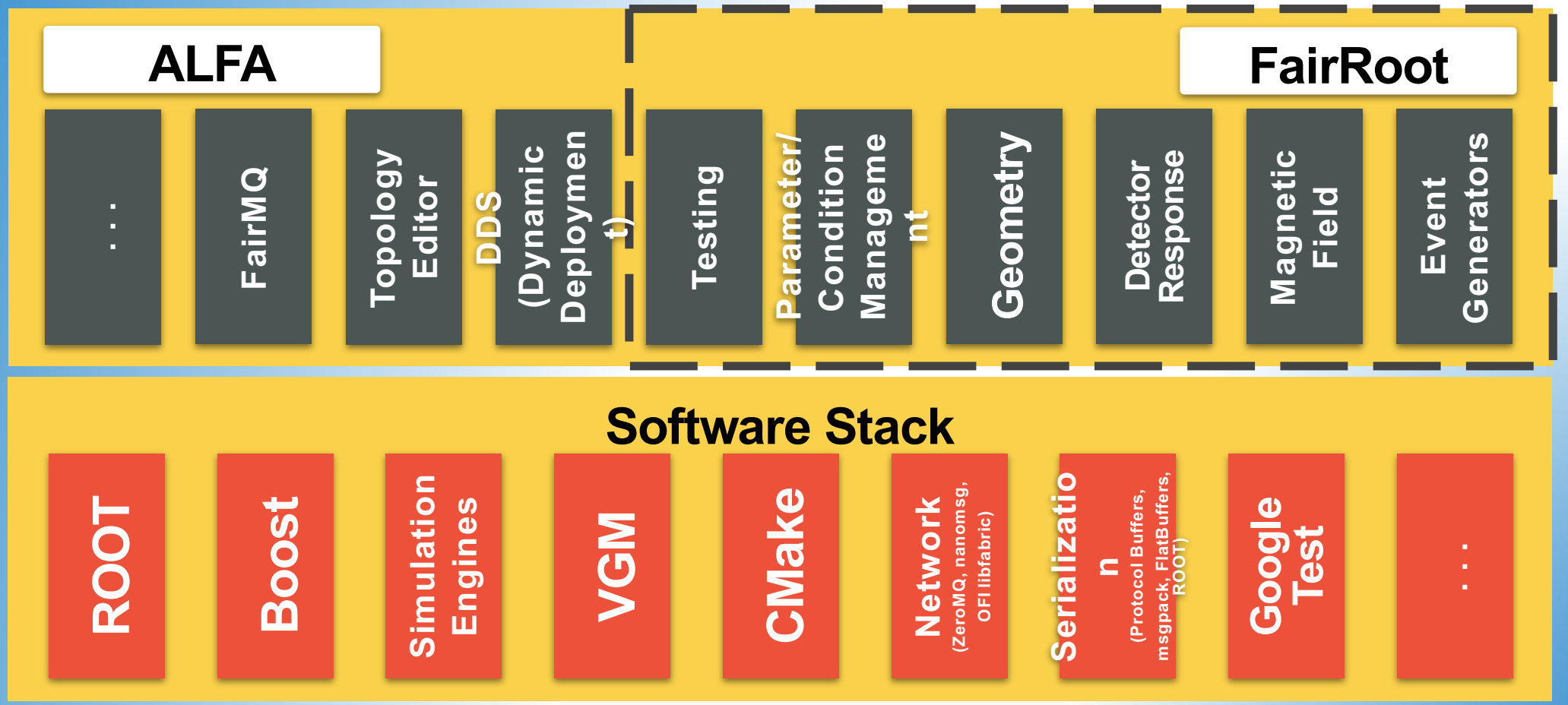
Summary, contd.

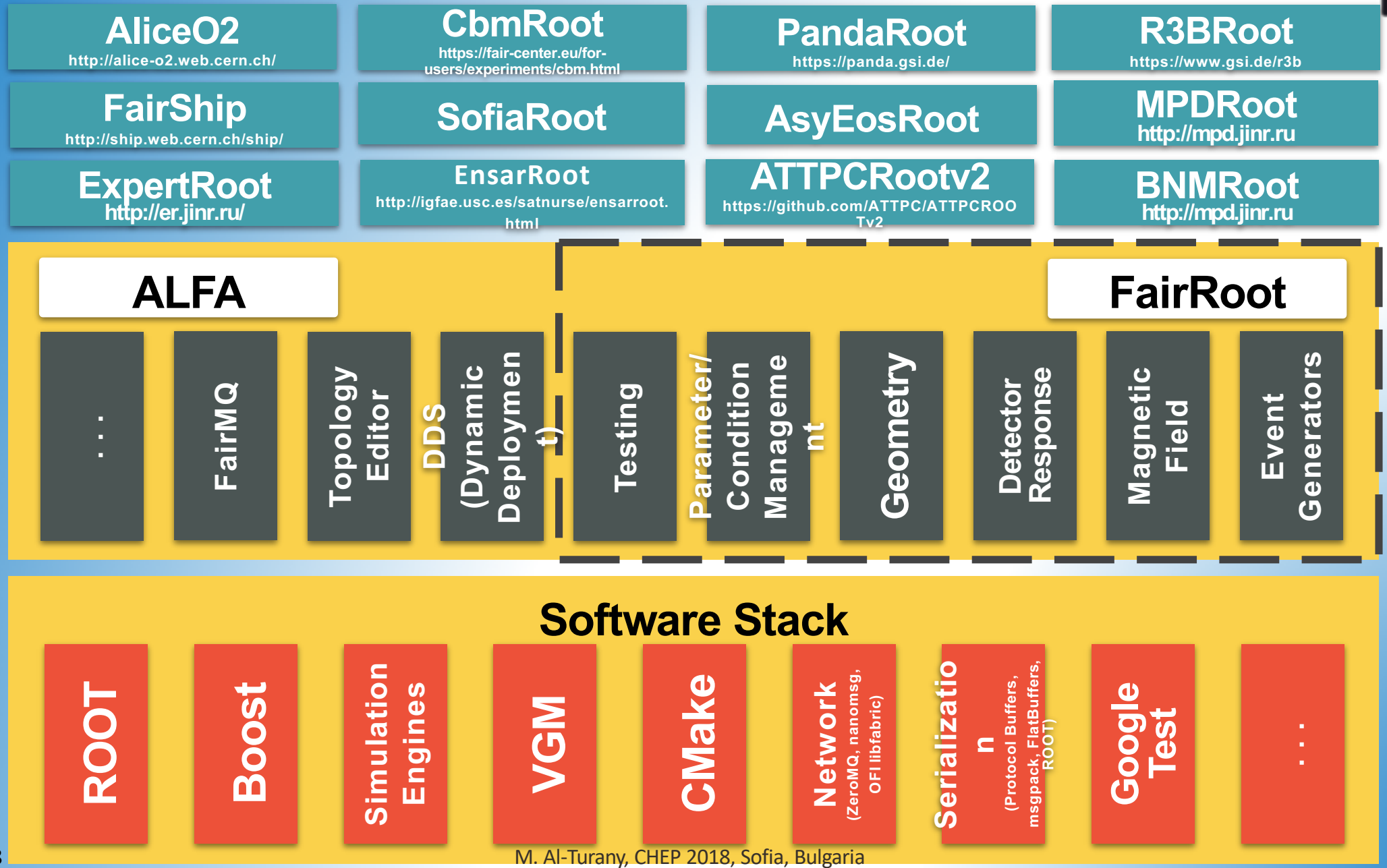
- Provides utilities to deploy topologies (via DDS) to computing clusters, online clusters as well as on a laptop
- Provides a message based Interface to access condition data (Parameter Manager)
- Has a plugin based system for configuration and control of user processes

https://github.com/FairRootGroup

Backup

What about FairRoot?




AliceO2
http://alice-o2.web.cern.ch/
CbmRoot
https://fair-center.eu/for-users/experiments/cbm.html
PandaRoot
https://panda.gsi.de/
R3BRoot
https://www.gsi.de/r3b
FairShip
http://ship.web.cern.ch/ship/
SofiaRoot
AsyEosRoot
MPDRoot
http://mpd.jinr.ru
ExpertRoot
http://er.jinr.ru/
EnsarRoot
http://igfae.usc.es/satnurse/ensarroot.html
ATTPCRootv2
https://github.com/ATTPC/ATTPCROOTv2
BNMRoot
http://mpd.jinr.ru
ALFA
FairRoot
. . .
FairMQ
Topology Editor
DDS (Dynamic Deployment)
Testing
Parameter/ Condition Management
Geometry
Detector Response
Magnetic Field
Event Generators
Software Stack
ROOT
Boost
Simulation Engines
VGM
CMake
Network (ZeroMQ, nanomsg, OFI libfabric)
Serialization (Protocol Buffers, msgpack, FlatBuffers, ROOT)
Google Test
. . .