

Anna Belova, JINR, 2020.

Prospects to use the FairMQ data exchange system for SPD

SPD ROOT

Monte Carlo simulation, event reconstruction for both simulated and real data, data analysis and visualization are planned to be performed by an object oriented C++ toolkit SPDroot. It is based on the FairRoot framework initially developed for the FAIR experiments at GSI Darmstadt and partially compatible with MPDroot and BM@Nroot software used at MPD and BM@N, respectively.

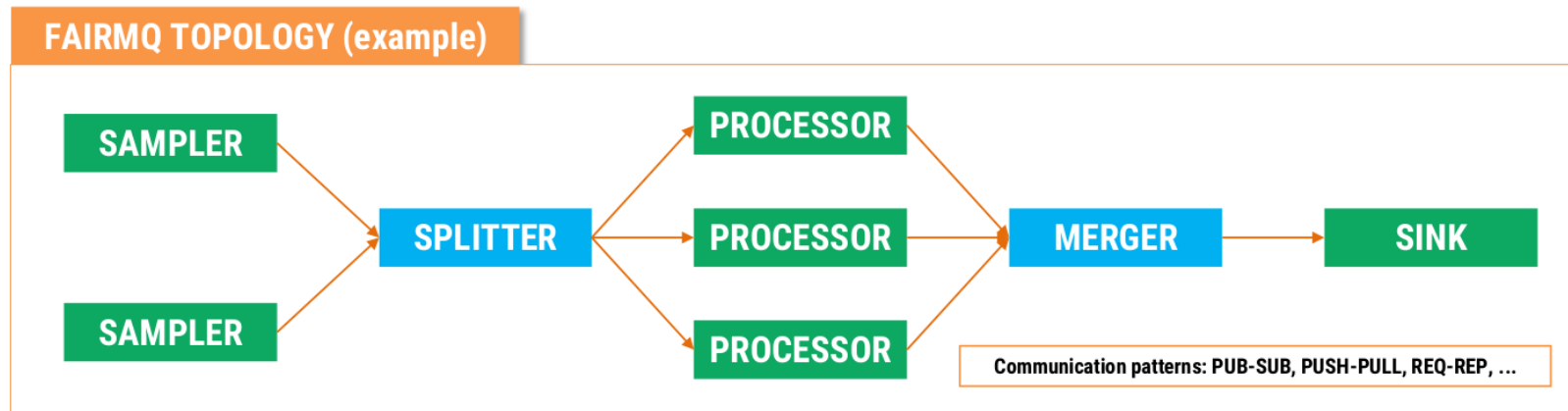
The SPD detector description for Monte Carlo simulation is based on the ROOT geometry while transportation of secondary particles through material of the setup and simulation of detector response is provided by GEANT4 code. The standard multipurpose generators like Pythia6 and Pythia8 as well as specialised generators can be used for simulation of primary nucleon-nucleon collision.

What is FairMQ

What is FairMQ?

Organize processing tasks in **topologies**, consisting of independent processes (**Device**: that communicate via *asynchronous message queues* over **network** or **inter-process**.

Ethernet, InfiniBand (IP-over-IB)



Ready to use devices are provided for typical scenarios.

User-defined devices can be implemented by inheriting from FairMQDevice.

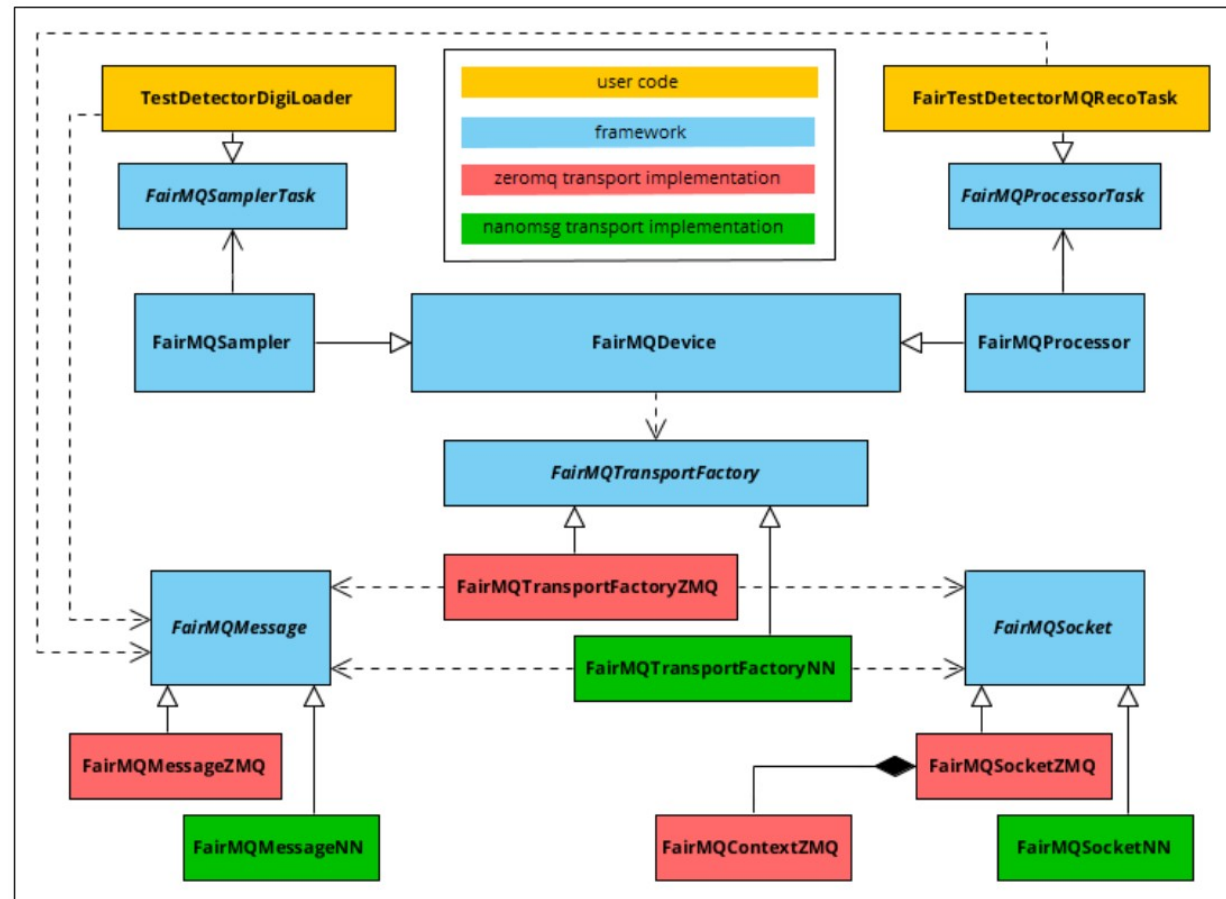
FairMQ structure

Transport Interface

FairMQ transport interface keeps the user code independent of the data transport implementation.

Currently two implementations:
With **ZeroMQ** or **nanomsg** libraries.

Possible implementation using future emerging technologies.



Parallelization throughput with FairMQ

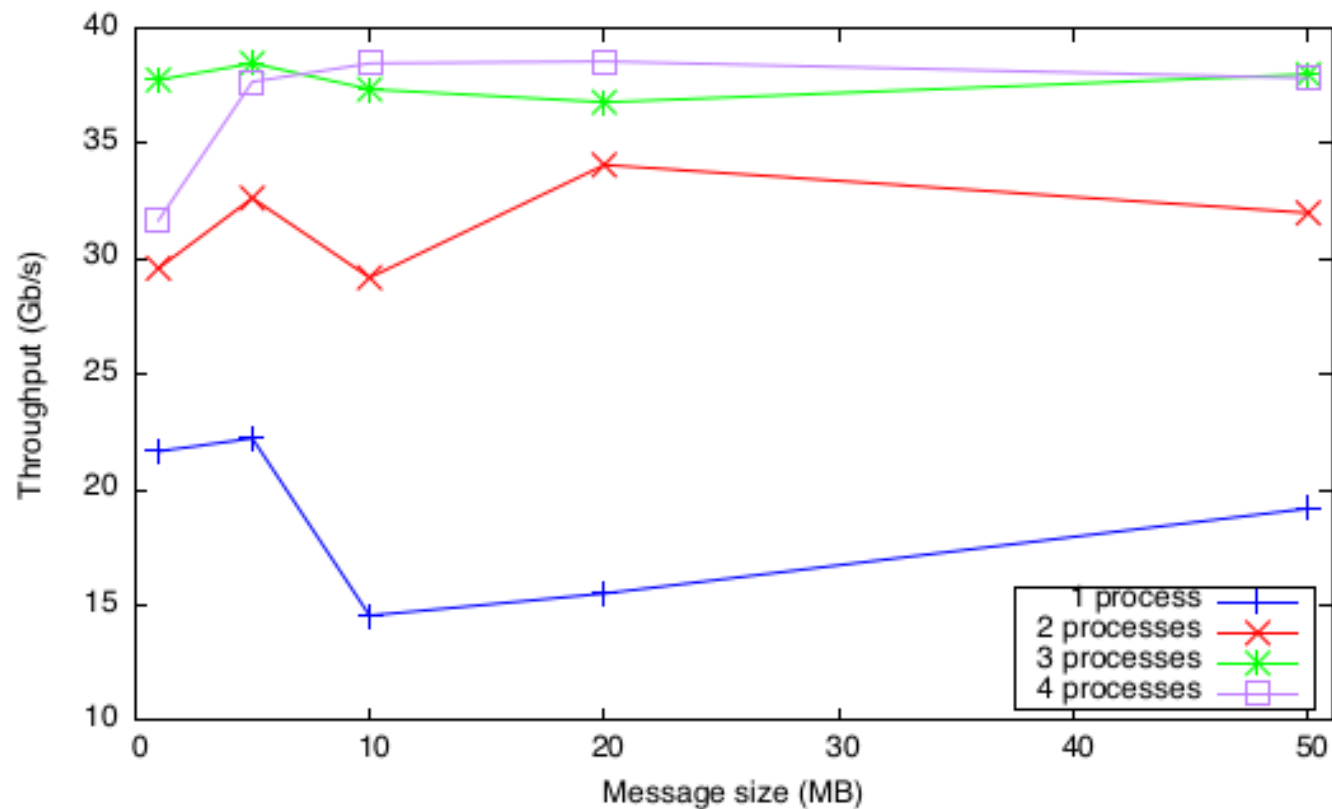


Figure 10.1: Throughput between two machines connected with 40 Gb/s Ethernet.

Data transport layer

The data transport layer is the part of the software which ensures the reliable arrival of message sand provides error checking mechanisms and data flow controls. The data transport layer in ALFA provides a number of components that can be connected to each other in order to construct a processing topology. They all share a common base class called device. Devices are grouped in three categories:

- Source: Devices without inputs are categorised as sources. A sampler is used to feed the pipeline (Task topology) with data from files.
- Message-based Processor: Devices that operate on messages without interpreting their content.
- Content-based Processor: This is the place where the message content is accessed and the user algorithms process the data.

Serialization

- Boost serialization. This method depends only on ANSI C++ facilities. Moreover, it exploits features of C++ such as RTTI (Run-Time Type Information), templates or multiple inheritance. It also provides independent versioning for each class definition. This means that when a class definition changes, older files can still be imported to the new version of the class. Another useful feature is the save and restore of deep pointers.
- Protocol buffers. Protocol buffers are Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data. The structure of the data is defined once and used to generate code to read and write data easily to and from a variety of data streams, using a variety of languages: Java, C++ or Python.
- ROOT. The ROOT Streamer can decompose ROOT objects into data members and write them to a buffer. This buffer can be written to a socket for sending over the network or to a file.
- User defined. In case it is decided not to use any of the above methods, binary structures or arrays can still be written or sent to a buffer. Although this method does not include any overhead for size of the data, issues can occur and will need to be managed. These include: schema evolution, different hardware, different languages.

Producer-Consumer with FairMQ extending

```
[16:36:35][DEBUG] Inserting new device channel from config: data
[16:36:35][STATE] BINDING ---> BOUND
[16:36:35][DEBUG] Setting 'zeromq' as default transport for the device
[16:36:35][DEBUG] Adding 'zeromq' transport
[16:36:35][STATE] BOUND ---> CONNECTING
[16:36:35][DEBUG] Validating channel 'data[0]'... VALID
[16:36:35][DEBUG] Transport: Using ZeroMQ library, version: 4.3.1
[16:36:35][DEBUG] Initializing transport for channel data[0]: default
[16:36:35][DEBUG] Reusing existing 'zeromq' transport
[16:36:35][STATE] INITIALIZING_DEVICE ---> INITIALIZED
[16:36:35][STATE] INITIALIZED ---> BINDING
[16:36:35][DEBUG] Validating channel 'data[0]'... VALID
[16:36:35][DEBUG] Created socket sink1.data[0].pull
[16:36:35][DEBUG] Created socket sampler1.data[0].push
[16:36:35][DEBUG] Attached channel data[0] to tcp://*:22184 (bind) (push)
[16:36:35][STATE] BINDING ---> BOUND
[16:36:35][STATE] BOUND ---> CONNECTING
[16:36:35][STATE] CONNECTING ---> DEVICE_READY
[16:36:35][STATE] DEVICE_READY ---> INITIALIZING_TASK
[16:36:35][STATE] INITIALIZING_TASK ---> READY
[16:36:35][DEBUG] Attached channel data[0] to tcp://127.0.0.1:22184 (connect) (pull)
[16:36:35][STATE] READY ---> RUNNING
[16:36:35][INFO] DEVICE: Running...
[16:36:35][STATE] CONNECTING ---> DEVICE_READY
[16:36:35][INFO] Sending "Hello"
[16:36:35][STATE] DEVICE_READY ---> INITIALIZING_TASK
[16:36:35][STATE] INITIALIZING_TASK ---> READY
[16:36:35][STATE] READY ---> RUNNING
[16:36:35][INFO] DEVICE: Running...
[16:36:35][INFO] Configured maximum number of iterations reached. Leaving RUNNING state.
[16:36:35][STATE] RUNNING ---> READY
[16:36:35][STATE] READY ---> RESETTING_TASK
[16:36:35][STATE] RESETTING_TASK ---> DEVICE_READY
[16:36:35][STATE] DEVICE_READY ---> RESETTING_DEVICE
[16:36:35][INFO] Received: "Hello"
[16:36:35][STATE] RESETTING_DEVICE ---> IDLE
[16:36:35][INFO] Configured maximum number of iterations reached. Leaving RUNNING state.
[16:36:35][STATE] IDLE ---> EXITING
[16:36:35][DEBUG] Shutting down Plugin Manager
[16:36:35][STATE] RUNNING ---> READY
[16:36:35][STATE] READY ---> RESETTING_TASK
[16:36:35][STATE] RESETTING_TASK ---> DEVICE_READY
[16:36:35][STATE] DEVICE_READY ---> RESETTING_DEVICE
[16:36:35][STATE] RESETTING_DEVICE ---> IDLE
[16:36:35][STATE] IDLE ---> EXITING
[16:36:35][DEBUG] Shutting down Plugin Manager
[16:36:35][DEBUG] Unloaded plugin: 'control', version '1.4.3', maintainer 'FairRoot'
[16:36:35][DEBUG] Shutting down Plugin Services
[16:36:35][DEBUG] Shutting down device sink1
[16:36:35][STATE] Exiting FairMQ state machine
[16:36:35][DEBUG] Unloaded plugin: 'control', version '1.4.3', maintainer 'FairRoot'
[16:36:35][DEBUG] Shutting down Plugin Services
[16:36:35][DEBUG] Shutting down device sampler1
[16:36:35][STATE] Exiting FairMQ state machine
anna@anna-System-Product-Name:~/fairsoft_jun19p2/basics/FairMQ/build/examples/1-1$
```

```
#!/bin/bash

export FAIRMQ_PATH=/home/anna/fairsoft_jun19p2/basics/FairMQ/build/fairmq
export LD_LIBRARY_PATH=.

transport="zeromq"

if [[ $1 =~ ^[a-z]+$ ]]; then
    transport=$1
fi

SESSION="$(/home/anna/fairsoft_jun19p2/basics/FairMQ/build/fairmq/fairmq-uuid-gen -h)"

# setup a trap to kill everything if the test fails/timeouts
trap 'kill -TERM $$SAMPLER_PID; kill -TERM $$SINK_PID; wait $$SAMPLER_PID; wait $$SINK_PID;' TERM

SAMPLER="fairmq-ex-1-1-sampler"
SAMPLER+= " --id sampler1"
SAMPLER+= " --rate 1"
SAMPLER+= " --transport $transport"
#SAMPLER+= " --verbosity veryhigh"
SAMPLER+= " --session $SESSION"
SAMPLER+= " --control static --color false"
SAMPLER+= " --max-iterations 1"
SAMPLER+= " --channel-config name=data,type=push,method=bind,address=tcp://*:22184,rateLogging=0"
/home/anna/fairsoft_jun19p2/basics/FairMQ/build/examples/1-1/$SAMPLER &
SAMPLER_PID=$!

SINK="fairmq-ex-1-1-sink"
SINK+= " --id sink1"
SINK+= " --transport $transport"
#SINK+= " --verbosity veryhigh"
SINK+= " --session $SESSION"
SINK+= " --control static --color false"
SINK+= " --max-iterations 1"
SINK+= " --channel-config name=data,type=pull,method=connect,address=tcp://localhost:22184,rateL"
/home/anna/fairsoft_jun19p2/basics/FairMQ/build/examples/1-1/$SINK &
SINK_PID=$!

# wait for sampler and sink to finish
wait $$SAMPLER_PID
wait $$SINK_PID
```

Producer

FairMQ channel

Consumer



```
17 FairMQDevicePtr getDevice(const FairMQProgOptions& config);
18
19 // to be implemented by the user to add custom command line options (or just with empty body
20 void addCustomOptions(boost::program_options::options_description&);
21
22 int main(int argc, char* argv[])
23 {
24     using namespace fair::mq;
25     using namespace fair::mq::hooks;
26
27     try
28     {
29         fair::mq::DeviceRunner runner(argc, argv);
30
31         // runner.AddHookLoadPlugins([](DeviceRunner& r){
32         //     // for example:
33         //     r.fPluginManager->SetSearchPaths({"Lib", "/Lib/plugins"});
34         //     r.fPluginManager->LoadPlugin("asdf");
35         // });
36
37         runner.AddHookSetCustomCmdLineOptions([](DeviceRunner& r){
38             boost::program_options::options_description customOptions("Custom options");
39             addCustomOptions(customOptions);
40             r.fConfig.AddToCmdLineOptions(customOptions);
41         });
42
43         // runner.AddHookModifyRawCmdLineArgs([](DeviceRunner& r){
44         //     // for example:
45         //     r.fRawCmdLineArgs.push_back("--blubb");
46         // });
47
48         runner.AddHookInstantiatedDevice([](DeviceRunner& r){
49             r.fDevice = std::unique_ptr<FairMQDevice>{getDevice(r.fConfig)};
50         });
51
52         return runner.Run();
53
54         // Run with builtin catch all exception handler, just:
55         // return runner.RunWithExceptionHandlers();
56     }
57     catch (std::exception& e)
58     {
59         LOG(error) << "Uncaught exception reached the top of main: " << e.what();
60         return 1;
61     }
62     catch (...)
63     {
64         LOG(error) << "Uncaught exception reached the top of main.";
65         return 1;
66     }
67 }
```


SimTask prototype

Running the simTask messaging prototype in spd_alfa:

1. Start parmqs-server with the following parameters: `--transport zeromq --id sim-parmq-server --channel-config name = updateChannel, type = rep, method = bind, rateLogging = 1, address = tcp: // * : 5205 --severity info --verbosity medium --color true --update-channel-name updateChannel --output-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.par`
`s.root` (correct the path to the last root file);
2. Run sink with the following parameters: `--transport zeromq --in-channel data # all # --id sim-sink1 --channel-config name = data # all #, type = pull, method = bind, rateLogging = 1, address = tcp: // *: 5206 --class-name FairMCEventHeader --branch-name MCEventHeader. --class-name TClonesArray --file-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.dat`
`a.root` (correct the path to the last root file - this will be the output file) ;
3. Run run-sim with the following parameters: `--transport zeromq --channel-config name = updateChannel, type = req, method = connect, rateLogging = 1, address = tcp: // localhost: 5205 --channel-config name = data # all #, type = push, method = connect, rateLogging = 1, address = tcp: // localhost: 5206 --severity info --verbosity medium --color true --nof-events 100 TGeant3 --id sim -sampler0 --random-seed 5` (the last number is random - you can change it)

Notes. The executable files are located in the folder where the build was made, and more specifically in `bin / anna_belova_prototype / MQ / simTask /`. All three applications need to run in three different terminals. The server and sync must continuously display status information. If this does not happen, then this is a failure, and you need to restart the application (this is very rare). Sometimes the tcp address is buggy - in this case, you need to change it. Running two servers at the same time at the same address will not work.

Start run with parmq-server and sink

```
[01:18:26][INFO]
airMQ version 1.4.3
      build RelWithDebInfo
      https://github.com/FairRootGroup/FairMQ
      LGPL-3.0 © 2012-2018 GSI

[01:18:26][STATE] Starting FairMQ state machine --> IDLE
[01:18:26][STATE] IDLE --> INITIALIZING_DEVICE
[01:18:26][STATE] INITIALIZING_DEVICE --> INITIALIZED
[01:18:26][STATE] INITIALIZED --> BINDING
[01:18:26][STATE] BINDING --> BOUND
[01:18:26][STATE] BOUND --> CONNECTING
[01:18:26][STATE] CONNECTING --> DEVICE_READY
[01:18:26][STATE] DEVICE_READY --> INITIALIZING_TASK
[01:18:26][STATE] INITIALIZING_TASK --> READY
[01:18:26][STATE] READY --> RUNNING
[01:18:26][INFO] DEVICE: Running...
Following control commands are available:
[h] help, [c] check current device state,
[t] init device, [b] bind, [x] connect, [j] init task, [r] run, [s] stop,
[r] reset task, [d] reset device, [q] end,
[k] increase log severity [l] decrease log severity [n] increase log verbosity
[m] decrease log verbosity

[01:18:27][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:28][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:29][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:30][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:31][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:32][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:33][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:34][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:35][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:36][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:37][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:38][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:39][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:40][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:41][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:42][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:43][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:44][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:18:45][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)

[01:18:39][INFO]
airMQ version 1.4.3
      build RelWithDebInfo
      https://github.com/FairRootGroup/FairMQ
      LGPL-3.0 © 2012-2018 GSI

[01:18:39][STATE] Starting FairMQ state machine --> IDLE
[01:18:39][DEBUG] PID: 4931
[01:18:39][DEBUG] Loaded plugin: 'control', version '1.4.3', maintainer 'FairRootGroup <fairroot@gsi.de>', homepage 'https://github.com/FairRootGroup/FairRoot/'
[01:18:39][DEBUG] Running builtin controller: interactive
[01:18:39][DEBUG] Plugin 'control' is setting up signal handling for SIGINT and SIGTERM
[01:18:39][STATE] IDLE --> INITIALIZING_DEVICE
[01:18:39][WARN] Config has no key: ack-channel. Returning default constructed object.
[01:18:40][INFO] SHOULD CREATE THE FILE AND TREE
[01:18:40][INFO] Creating output branch "FairMCEventHeader" with name "MCEventHeader."
[01:18:40][DEBUG] Inserting new device channel from config: data#all#
[01:18:40][DEBUG] Setting 'zeromq' as default transport for the device
[01:18:40][DEBUG] Adding 'zeromq' transport
[01:18:40][DEBUG] Transport: Using ZeroMQ library, version: 4.3.1
[01:18:40][DEBUG] Initializing transport for channel data#all#[0]: default
[01:18:40][DEBUG] Reusing existing 'zeromq' transport
[01:18:40][STATE] INITIALIZING_DEVICE --> INITIALIZED
[01:18:40][STATE] INITIALIZED --> BINDING
[01:18:40][DEBUG] Validating channel 'data#all#[0]... VALID
[01:18:40][DEBUG] Created socket sim-sink1.data#all#[0].pull
[01:18:40][DEBUG] Attached channel data#all#[0] to tcp://*:5206 (bind) (pull)
[01:18:40][STATE] BINDING --> BOUND
[01:18:40][STATE] BOUND --> CONNECTING
[01:18:40][STATE] CONNECTING --> DEVICE_READY
[01:18:40][STATE] DEVICE_READY --> INITIALIZING_TASK
[01:18:40][STATE] INITIALIZING_TASK --> READY
[01:18:40][STATE] READY --> RUNNING
[01:18:40][INFO] DEVICE: Running...
Following control commands are available:
[h] help, [c] check current device state,
[t] init device, [b] bind, [x] connect, [j] init task, [r] run, [s] stop,
[r] reset task, [d] reset device, [q] end,
[k] increase log severity [l] decrease log severity [n] increase log verbosity
[m] decrease log verbosity

[01:18:41][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
```

Start run with run-sim

```
Обзор Терминал Вт, 01:37
Терминал
Файл Правка Вид Поиск Терминал Справка
[01:33:39][INFO]
FairMQ version 1.4.3
build RelWithDebInfo
https://github.com/FairRootGroup/FairMQ
LGPL-3.0 © 2012-2018 GSI
0)!!!
[01:33:39][STATE] Starting FairMQ state machine --> IDLE
0a)!!!
0b)!!!
0c)!!!
[01:33:39][INFO] Media file used: /home/bel/work/spd_alfa/geometry/media.geo
0d)!!!
0e)!!!
0f)!!!
-I- <SpdCommonGeoMapper::DefineQslGeometrySet>
0i)!!!
0j)!!!
1)!!!
-I- <SpdCommonGeoMapper::AddPassive> Pipe
-I- <SpdCommonGeoMapper::AddPassive> Magnet (hyb)
2)!!!
-I- <SpdCommonGeoMapper::AddDetector> Ecal barrel (tor)
-I- <SpdCommonGeoMapper::AddDetector> Ecal endcaps (tor)
-I- <SpdCommonGeoMapper::AddDetector> RS barrel (qsl)
-I- <SpdCommonGeoMapper::AddDetector> RS endcaps (qsl)
-I- <SpdCommonGeoMapper::AddDetector> TS barrel (tor)
-I- <SpdCommonGeoMapper::AddDetector> TS endcaps (tor)
-I- <SpdCommonGeoMapper::AddDetector> Inner tracker system
3)!!!
-I- <SpdTsTECGeoMapper::SetParameters> module/layer/straw: 1/1/1
-I- <SpdTsTECGeoMapper::SetParameters> module/layer/straw: 2/2/1
4)!!!
-I- <SpdAxialFieldMap::InitData> Path to the field map ($MAGFPATH): /home/bel/work/spd_alfa/anna_belova_prototype/input
-I- <SpdAxialFieldMapData::ReadBinaryFile> Reading file: /home/bel/work/spd_alfa/anna_belova_prototype/input/map_qsolRZ_6cls2cm.bin
-I- <SpdAxialFieldMapData::ReadBinaryFile> Read field values: 10912/10912
-I- <SpdAxialFieldMap::InitData>
5)!!!
<SpdField::Print>
Field Name: QSolenoidal field
Field Type: 3 (AxialMap)
Region Type: tube
Region on/off: 1
<SpdAxialFieldMap::PrintFieldParameters>
Init level: 2
Approximation method: 0
M(Br) = 8.000000e-01
M(Bz) = 8.000000e-01
<SpdAxialFieldMapData::Print>
Data map name: QSolenoidal field
Path to data: /home/bel/work/spd_alfa/anna_belova_prototype/input/
file name: map_qsolRZ_6cls2cm.bin
```


Init run

```
Обзор Терминал Вт, 01:37
Терминал
Файл Правка Вид Поиск Терминал Справка

=====
I
I ISUB Subprocess name I Maximum value I
I
I
I
I 95 Low-pT scattering I 2.3474D+01 I
I 96 Semihard QCD 2 -> 2 I 8.3517D+02 I
I
I
=====

***** PYMULT: initialization of multiple interactions for MSTP(82) = 4 *****
pT0 = 1.02 GeV gives sigma(parton-parton) = 2.15D+01 mb: rejected
pT0 = 0.92 GeV gives sigma(parton-parton) = 2.95D+01 mb: accepted

***** PYMIGN: initialization of multiple interactions for MSTP(82) = 4 *****
pT0 = 0.92 GeV gives sigma(parton-parton) = 3.10D+01 mb: accepted

***** PYINIT: initialization completed *****

7)!!!

+++++
+
+ Init Run (start) +
+
+++++

[01:33:39][STATE] IDLE ---> INITIALIZING_DEVICE
[01:33:39][STATE] INITIALIZING_DEVICE ---> INITIALIZED
[01:33:39][STATE] INITIALIZED ---> BINDING
[01:33:39][STATE] BINDING ---> BOUND
[01:33:39][STATE] BOUND ---> CONNECTING
[01:33:39][STATE] CONNECTING ---> DEVICE_READY
[01:33:39][STATE] DEVICE_READY ---> INITIALIZING_TASK
FairMQSimDevice::InitTask()-3
FairMQSimDevice::InitTask()-6
FairMQSimDevice::InitTask()-7: c_str=ReportSimDevice, const_cast char=ReportSimDevice, length=15
FairMQSimDevice::InitTask()-8
FairMQSimDevice::InitTask()-9
FairMQSimDevice::InitTask()-10
FairMQSimDevice::InitTask()-11
FairMQSimDevice::InitTask()-12
FairMQSimDevice::InitTask()-13
[01:33:39][INFO] -> 1610404419_0
FairMQSimDevice::InitTask()-14
FairMQSimDevice::InitTask()-15
FairMQSimDevice::InitTask()-16
FairMQSimDevice::InitTask()-17
[01:33:39][INFO] runId = 1610404419 /// fSimDeviceId = 0
FairMQSimDevice::InitTask()-18
FairMQSimDevice::InitTask()-19
FairMQSimDevice::InitTask()-20
FairMQSimDevice::InitTask()-21
FairMQSimDevice::InitTask()-24
```

Parmq-server and sink during run-sim starting

```
[01:23:15][INFO] got process update message with size = 8256 !
-I- <SpdCommonGeoMapper::AddPassive> Pipe
-I- <SpdCommonGeoMapper::AddPassive> Magnet (hyb)
Info in <TGeoManager::TGeoManager>: Geometry Geometry, default geometry created
-I- <SpdCommonGeoMapper::AddDetector> TS barrel (tor)
-I- <SpdCommonGeoMapper::AddDetector> Ecal barrel (tor)
-I- <SpdCommonGeoMapper::AddDetector> RS barrel (qsl)
-I- <SpdCommonGeoMapper::AddDetector> TS endcaps (tor)
-I- <SpdCommonGeoMapper::AddDetector> Ecal endcaps (tor)
-I- <SpdCommonGeoMapper::AddDetector> RS endcaps (qsl)
-I- <SpdCommonGeoMapper::AddDetector> Inner tracker system

*****
initialisation for run id 1610403789
*****
OBJ: FairBaseParSet      FairBaseParSet      class for parameter io
[01:23:16][INFO] *** FairBaseParSet written to ROOT file  version: 88
[01:23:16][INFO] got process update message with size = 21004288 !
Info in <TGeoManager::CloseGeometry>: Geometry loaded from file...
Info in <TGeoManager::SetTopVolume>: Top volume is cave. Master volume is cave
Info in <TGeoNavigator::BuildCache>: --- Maximum geometry depth set to 100
Info in <TGeoManager::Voxelize>: Voxelizing...
Info in <TGeoManager::CountLevels>: max level = 6, max placements = 335
Info in <TGeoManager::CloseGeometry>: 500482 nodes/ 1971 volume UID's in FAIR geo
omety
Info in <TGeoManager::CloseGeometry>: -----modeler ready-----
---

*****
initialisation for run id 1610403789
*****
-I- FairRunTimeDB::InitContainer() FairBaseParSet
OBJ: FairGeoParSet      FairGeoParSet      class for Geo parameter
[01:23:16][INFO] updateChanne[0]: in: 2 (21.0125 MB) out: 1 (7e-06 MB)
[01:23:16][INFO] *** FairGeoParSet written to ROOT file  version: 88
[01:23:16][INFO] got process update message with size = 2064 !

*****
initialisation for run id 1610403789
*****
-I- FairRunTimeDB::InitContainer() FairBaseParSet
-I- FairRunTimeDB::InitContainer() FairGeoParSet
OBJ: SpdFieldPar      SpdFieldPar      Spd Field Parameters
[01:23:16][INFO] *** SpdFieldPar written to ROOT file  version: 4
[01:23:16][INFO] got process update message with size = 1032 !

*****
initialisation for run id 1610403789
*****
-I- FairRunTimeDB::InitContainer() FairBaseParSet
-I- FairRunTimeDB::InitContainer() FairGeoParSet
-I- FairRunTimeDB::InitContainer() SpdFieldPar
OBJ: SpdPassiveGeoParSet      PassiveGeoParSet      Spd Passive Geo Paramete
```

```
[01:23:13][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:14][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:15][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:16][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:17][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:18][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:19][DEBUG] - RTDB container factory SpdFieldContFact
[01:23:19][DEBUG] - RTDB container factory SpdTsTContFact
[01:23:19][DEBUG] - RTDB container factory SpdEcalContFact
[01:23:19][DEBUG] - RTDB container factory SpdRsTContFact
[01:23:19][DEBUG] - RTDB container factory SpdItsContFact
[01:23:19][INFO] data#all#[0]: in: 1 (1.29722 MB) out: 0 (0 MB)
[01:23:20][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:21][INFO] data#all#[0]: in: 1 (1.25904 MB) out: 0 (0 MB)
[01:23:22][INFO] data#all#[0]: in: 1 (1.28897 MB) out: 0 (0 MB)
[01:23:23][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:24][INFO] data#all#[0]: in: 1 (1.35502 MB) out: 0 (0 MB)
[01:23:25][INFO] data#all#[0]: in: 1 (1.07431 MB) out: 0 (0 MB)
[01:23:26][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:27][INFO] data#all#[0]: in: 1 (1.0578 MB) out: 0 (0 MB)
[01:23:28][INFO] data#all#[0]: in: 1 (1.23118 MB) out: 0 (0 MB)
[01:23:29][INFO] data#all#[0]: in: 1 (1.15378 MB) out: 0 (0 MB)
[01:23:30][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:31][INFO] data#all#[0]: in: 1 (1.12694 MB) out: 0 (0 MB)
[01:23:32][INFO] data#all#[0]: in: 1 (1.31374 MB) out: 0 (0 MB)
[01:23:33][INFO] data#all#[0]: in: 1 (1.16513 MB) out: 0 (0 MB)
[01:23:34][INFO] data#all#[0]: in: 1 (0.916416 MB) out: 0 (0 MB)
[01:23:35][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:36][INFO] data#all#[0]: in: 1 (1.19093 MB) out: 0 (0 MB)
[01:23:37][INFO] data#all#[0]: in: 1 (1.29722 MB) out: 0 (0 MB)
[01:23:38][INFO] data#all#[0]: in: 1 (1.01652 MB) out: 0 (0 MB)
[01:23:39][INFO] data#all#[0]: in: 1 (1.20228 MB) out: 0 (0 MB)
[01:23:40][INFO] data#all#[0]: in: 1 (1.04232 MB) out: 0 (0 MB)
[01:23:41][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:42][INFO] data#all#[0]: in: 1 (1.12305 MB) out: 0 (0 MB)
[01:23:43][INFO] data#all#[0]: in: 1 (1.32199 MB) out: 0 (0 MB)
[01:23:44][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:45][INFO] data#all#[0]: in: 1 (1.16513 MB) out: 0 (0 MB)
[01:23:46][INFO] data#all#[0]: in: 1 (0.834888 MB) out: 0 (0 MB)
[01:23:47][INFO] data#all#[0]: in: 1 (1.22292 MB) out: 0 (0 MB)
[01:23:48][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:49][INFO] data#all#[0]: in: 1 (1.19712 MB) out: 0 (0 MB)
[01:23:50][INFO] data#all#[0]: in: 1 (1.29722 MB) out: 0 (0 MB)
[01:23:51][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:52][INFO] data#all#[0]: in: 1 (1.42106 MB) out: 0 (0 MB)
[01:23:53][INFO] data#all#[0]: in: 1 (1.18164 MB) out: 0 (0 MB)
[01:23:54][INFO] data#all#[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:23:55][INFO] data#all#[0]: in: 1 (1.04954 MB) out: 0 (0 MB)
```

Simulation finishing

The screenshot displays the Qt Creator IDE interface. The main window shows a terminal window with simulation logs. The logs consist of a series of 'updateChannel' and 'data#all' messages, each with 'in' and 'out' data flow information. The time stamps range from 01:35:36 to 01:36:00. Below the terminal, a 'Статический анализатор' (Static Analyzer) window is open, showing a table of components and their properties. The table includes columns for component name, ID, and various parameters. The total simulation time is 25.75 seconds, and the total memory usage is 25.85 MB. The bottom status bar shows the search bar and navigation tabs for '1 Проблемы', '2 Результаты поиска', '3 Вывод приложения', '4 Консоль сборки', '5 Консоль отладчика', '6 Основные сообщения', and '8 Результаты тестирования'.

```
[01:35:36][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:39][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:40][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:41][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:42][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:43][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:44][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:45][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:46][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:47][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:48][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:49][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:50][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:51][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:52][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:53][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:54][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:55][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:56][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:57][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:58][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:59][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:36:00][INFO] updateChannel[0]: in: 0 (0 MB) out: 0 (0 MB)

[01:35:37][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:38][INFO] data#all[0]: in: 1 (1.18164 MB) out: 0 (0 MB)
[01:35:39][INFO] data#all[0]: in: 1 (1.22705 MB) out: 0 (0 MB)
[01:35:40][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:41][INFO] data#all[0]: in: 1 (1.64398 MB) out: 0 (0 MB)
[01:35:42][INFO] data#all[0]: in: 1 (1.37359 MB) out: 0 (0 MB)
[01:35:43][INFO] data#all[0]: in: 1 (1.25594 MB) out: 0 (0 MB)
[01:35:44][INFO] data#all[0]: in: 1 (1.16513 MB) out: 0 (0 MB)
[01:35:45][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:46][INFO] data#all[0]: in: 1 (1.56142 MB) out: 0 (0 MB)
[01:35:47][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:48][INFO] data#all[0]: in: 1 (1.20641 MB) out: 0 (0 MB)
[01:35:49][INFO] data#all[0]: in: 1 (1.29722 MB) out: 0 (0 MB)
[01:35:50][INFO] data#all[0]: in: 1 (0.983496 MB) out: 0 (0 MB)
[01:35:51][INFO] data#all[0]: in: 2 (1.44686 MB) out: 0 (0 MB)
[01:35:52][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:53][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:54][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:55][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:56][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:57][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:58][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
[01:35:59][INFO] data#all[0]: in: 0 (0 MB) out: 0 (0 MB)
```

| Идентификатор | Имя компонента | Параметры | Время (с) | Память (МБ) |
|---------------|-------------------|------------------------|-----------|-------------|
| 5 | RS endcaps (qsl) | | 0.00 | 0.00 |
| 0000 | RsTEC | 0.001205 | | |
| 6 | Ecal barrel (tor) | | 16.61 | |
| 0237 | EcalTB | 81722.365552, 4.920000 | | |
| 7 | SpdEcalTEC | Ecal endcaps (tor) | 9.14 | |
| 5386 | EcalTEC | 44995.299702, 4.920000 | | |
| 8 | SpdTsTB | TS barrel (tor) | 0.00 | |
| 0000 | TsTB | 0.000000 | | |
| 9 | SpdTSTEC | TS endcaps (tor) | 0.00 | |
| 0000 | TsTEC | 0.000000 | | |
| total: | | | | 25.75 |
| TOTAL: | | | | 25.85 |

```
FairMQSimDevice::Run()-20
WARNING - Attempt to delete the physical volume store while geometry closed !
WARNING - Attempt to delete the logical volume store while geometry closed !
WARNING - Attempt to delete the solid store while geometry closed !
WARNING - Attempt to delete the region store while geometry closed !
Для закрытия данного окна нажмите <ВВОД>...
```

DDS

The Dynamic Deployment System (DDS) is an independent set of utilities and interfaces, providing a dynamic distribution of different user processes for any given topology on any Resource Management System (RMS). The DDS uses a plug-in system in order to deploy different job submission front-ends. The first and the main plug-in of the system is a Secure Shell (SSH) that can be used to dynamically transform a set of machines into user worker nodes. The DDS functions are the following:

- Deploy a task or set of tasks
- Use any RMS (Slurm, Grid Engine, ...etc)
- Execute nodes securely (watchdog)
- Support different topologies and task dependencies
- Support a central log engine

During 2014, the core modules of the DDS were developed and the first stable prototype was released. This has been tested on the ALICE HLT development cluster using 40 computing nodes with 32 processes per node. The SSH plugin for DDS has been used to successfully distribute and manage 1281 ALICE O2 user tasks (640 First Level Processor (FLPs) and 640 Event Processing units (EPN)). The FLP processes here are emulating the FLP nodes which will collect the data whereas the EPN emulates the second step of data processing: assigning each cluster to a track ([10]) The DDS was able to propagate the allocated ports for each process to the dependent processes and set the required topology for the test. Throughout the test on this cluster, one DDS commander server propagated more than 1.5 million properties in less than 5 seconds.

DDS commands

DDS command line interface is simple and user friendly. A quick start can look like the following:

```
$ dds-session start
```

```
$ dds-submit-r ssh-c hosts.cfg
```

```
$ dds-topology--activate topology.xml
```

```
$ dds-topology--update new-topology.xml
```

The first command starts a DDS session with a commander server. The second one submits DDS scouts which spawn DDS agents for the current session and act as a watchdog for them. In the given example we use SSH plug-in together with the configuration file. After the agents are submitted we are ready to activate the topology. The third command does that. If required the running topology can be updated with the new one without stopping using the fourth command.

Topology for SimuSql

```
<topology id="SimuSqlDDS">

  <property id="data#all#" />
  <property id="updateChannel" />

  <decltask id="Sampler">
    <exe reachable="true">/home/bel/work/spd_alfa/build/bin/anna_belova_prototype/MQ/simTask/run-sim --transport zeromq --channel-config name=updateChannel,type=req,method=connect,rateLogging=1,address=tcp://localhost:5205 --channel-config name=data#all#,type=push,method=connect,rateLogging=1,address=tcp://localhost:5206 --severity info --verbosity medium --color true --nof-events 100 TGeant3 --id sim-sampler0 --random-seed 5</exe>
    <properties>
      <id access="read">data#all#</id>
    <id access="read">updateChannel</id>
  </properties>
</decltask>

  <decltask id="Sink">
    <exe reachable="true">/home/bel/work/spd_alfa/build/bin/anna_belova_prototype/MQ/simTask/sink --transport zeromq --in-channel data#all# --id sim-sink1 --channel-config name=data#all#,type=pull,method=bind,rateLogging=1,address=tcp://*:5206 --class-name FairMCEventHeader --branch-name MCEventHeader. --class-name TClonesArray --file-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.data.root</exe>
    <properties>
      <id access="write">data#all#</id>
    </properties>
  </decltask>

  <decltask id="ParamServer">
    <exe reachable="true">/home/bel/work/spd_alfa/build/bin/parmq-server --transport zeromq --id sim-parmq-server --channel-config name=updateChannel,type=rep,method=bind,rateLogging=1,address=tcp://*:5205 --severity info --verbosity medium --color true --update-channel-name updateChannel --output-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.pars.root</exe>
    <properties>
      <id access="write">updateChannel</id>
    </properties>
  </decltask>

  <main id="main">
    <task>ParamServer</task>
    <task>Sink</task>
    <task>Sampler</task>
  </main>

</topology>
```

Topology activation

```
Обзор Терминал BT, 21:43 en
bel@Bel-PC: ~

Файл Правка Вид Поиск Терминал Справка
dds-submit: Server reports: Plug-in: Checking status of agents...
dds-submit: Server reports: Plug-in: All agents have been started successfully
dds-submit: Server reports: Plug-in: Validating...
dds-submit: Server reports: Plug-in: All agents have been validated successfully
bel@Bel-PC:~$ dds-topology --activate /home/bel/work/FairRoot/examples/MQ/pixelDetector/run/scripts/simuSql-dds.xml
dds-topology: Contacting DDS commander on Bel-PC:20001 ...
dds-topology: Connection established.
dds-topology: Requesting server to activate/update/stop a topology...
dds-topology: Activating topology /home/bel/work/FairRoot/examples/MQ/pixelDetector/run/scripts/simuSql-dds.xml
dds-topology:
Removed tasks: 0
Removed collections:0
Added tasks :3
1 x main/ParamServer
1 x main/Sampler
1 x main/Sink
Added collections: 0

dds-topology: Updating topology for agents...
[=====] 100 % (3/3)
Updated agent topologies: 3
Errors: 0
Total: 3
Time to update agent topologies: 0.002 s
dds-topology: Uploading user tasks...
[=====] 100 % (3/3)
Assigned/Uploaded tasks: 3
Errors: 0
Total: 3
Time to assign/upload: 0.394 s
dds-topology: Assigning user tasks...
[=====] 100 % (3/3)
Assigned/Uploaded tasks: 3
Errors: 0
Total: 3
Time to assign/upload: 0 s
dds-topology: Activating user tasks...
[=====] 100 % (3/3)
Activated tasks: 3
Errors: 0
Total: 3
Time to Activate: 0.001 s
bel@Bel-PC:~$ ps xawu | grep parm-q-server
bel  2227 96.6  0.1 402020 31012 ?        Sl  21:42   0:00 /home/bel/work/spd_alfa/build/bin/parm-q-server --transport zeromq --id sim-parm-q-server --channel-config name=updateChannel,type=rep,method=bind,rateLogging=1,address=tcp://*:5205 --severity info --verbosity medium --color true --update-channel-name updateChannel --output-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.pars.root
bel  2273  0.0  0.0 15656 1060 pts/0    S+  21:42   0:00 grep --color=auto parm-q-server
bel@Bel-PC:~$ ps xawu | grep sink
bel  2225 91.2  0.9 620224 160308 ?        Sl  21:42   0:13 /home/bel/work/spd_alfa/build/bin/anna_belova_prototype/MQ/simTask/sink --transport zeromq --in-channel data#all# --id sim-sink1 --channel-config name=data#all#,type=pull,method=bind,rateLogging=1,address=tcp://*:5206 --class-name FairMCEventHeader --branch-name MCEventHeader. --class-name TClonesArray --file-name /home/bel/work/spd_alfa/anna_belova_prototype/MQ/simTask/macros/MQ.simulation_TGeant3.data.root
bel  2275  0.0  0.0 15656 1012 pts/0    S+  21:43   0:00 grep --color=auto sink
bel@Bel-PC:~$ ps xawu | grep run-sim
bel  2277  0.0  0.0 15656 1056 pts/0    S+  21:43   0:00 grep --color=auto run-sim
bel@Bel-PC:~$
```

Conclusions

FairMQ uses ZeroMQ as its main transport layout and therefore has superior process parallelization, data integrity, and easy multithreading capabilities. ALICE O2 experiments have demonstrated high throughput using FairMQ, and therefore, there are good prospects for using the FairMQ package in SPD experiments.

References

1. <http://spd.jinr.ru/spd-software/>
2. Alexey Rybalchenko, GSI Darmstadt, FairRoot group, FairMQ Data Transport for Online & Offline Processing, ALICE Offline Week CERN, July 1, 2015
3. M. Al-Turany^{1,2}, P. Buncic², P. Hristov², T. Kollegger¹, C.Kouzinopoulos², A. Lebedev¹, V. Lindenstruth^{1,3}, A. Manafov¹, M.Richter^{2,4}, A. Rybalchenko¹, P. Vande Vyvre², N. Winckler: ALFA: The new ALICE-FAIR software framework
4. <http://wiki.zeromq.org/intro:read-the-manual>

The end

**Thank you
for
attention!**