

02 TUTORIAL & FEEDBACK

Giulio Eulisse

O2 TUTORIAL

IDEA: an "O2 Primer" tutorial where the user is guided end-to-end to:

- Set up her work environment (using alibuild)
- Create a simple sampler - sink topology (slightly simplified version of ex1 in FairRoot)
- Deploy it using DDS (WIP)

Current draft at: <http://ktf.github.io/o2-tutorial/>

PERSONAL COMMENTS WHILE PREPARING THE TUTORIAL

FairROOT configs are not valid JSON

Nicola agrees and says it will be fixed.

FairROOT configs are not self contained

You need to have both the config and the name of each executable if you want to run a topology. Why not adding the name of the executable to the config itself? This could be then parsed by a simple driver process which takes care of instantiating every device in the topology.

Memory ownership in "CreateMessage(data, size, destructor, hint)"

I would suggest to make "destructor" (ffn) obligatory, rather than optional like now. This would make sure that people who use this API actually know what they are doing (and to some extent discourage its usage for the general case as discussed last time).

PERSONAL COMMENTS WHILE PREPARING THE TUTORIAL

SimpleO2App wrapper?

I was also rambling about a wrapper class which hides most of the details of the state machine and configuration management. Maybe what I am really after is using FairMQProcessor / FairMQSampler / FairMQSink and the task based API.

CMake setup

Personally, I would stick to pure CMake when possible and avoid for loops in CMakeFiles to save a couple of lines of code, at least in examples.

Too many environment variables which can affect compilation (I think Florian and Mohammad already agree this is to be cleaned up).

How do we plan to keep dependencies of user packages under control?

We are planning to meet with Matthias, Barth, Mohammad (and whoever else is interested) to discuss a bit what can be done to cleanup the situation.