

## artdaq - Idea #22529

### Support the (proto)DUNE DFO Model in artdaq

05/07/2019 09:14 PM - Eric Flumerfelt

<b>Status:</b> New	<b>Start date:</b> 05/07/2019
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b>	<b>% Done:</b> 14%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b>	
<b>Experiment:</b> -	
<b>Description</b>	
I'm opening this as a meta-issue to contain the various (and future) pieces of work we have identified as necessary for providing functionality equivalent to the DFO requirement for DUNE.	
I have attached images from the meeting today with Kurt, Wes and myself.	
<b>Subtasks:</b>	
Feature # 22530: Separate Token Reception from RoutingMasterCore into a subclass	<b>Closed</b>
Feature # 22531: Add destination parameter to DSM::sendFragment	<b>Assigned</b>
Feature # 22532: Add acknowledgements to Request protocol	<b>Resolved</b>
Feature # 22533: Change signature of CommandableFragmentGenerator::getNext	<b>Assigned</b>
Feature # 22534: Bookkeeping changes for allowing multiple request domains within a sub...	<b>New</b>
Feature # 22535: RoutingNetOutput plugin: A version of BinaryNetOutput that receives Ro...	<b>New</b>
Feature # 22569: Update RequestReceiver's interface	<b>Resolved</b>

#### History

##### #1 - 05/07/2019 09:15 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22530](#)

##### #2 - 05/07/2019 09:17 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22531](#)

##### #3 - 05/07/2019 09:19 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22532](#)

##### #4 - 05/07/2019 09:21 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22533](#)

##### #5 - 05/07/2019 09:23 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22534](#)

##### #6 - 05/07/2019 09:26 PM - Eric Flumerfelt

- Due date set to 05/07/2019

due to changes in a related task: [#22535](#)

**#7 - 05/09/2019 12:27 PM - Kurt Biery**

As an initial step toward returning destination information, along with the `artdaq::Fragments`, from `CommandableFragmentGenerator::getNext()`, would it be OK to update the `RequestReceiver` interface to add a method that returns the destination rank for a specified sequence ID?

Another way to phrase this question: do we want to modify the existing `RequestReceiver` interface to return destination information, along with sequence ID and timestamp information, from the `GetRequests()` and `GetAndClearRequests()` methods,  
OR

do we want to keep the existing interface signatures and add a new method to fetch the destination for a specified sequence ID?

**#8 - 05/09/2019 12:36 PM - Eric Flumerfelt**

This is a bit trickier than I realized, since the rank is in the header for the request block...For requests from the `RoutingMaster`, we may want the rank to be in the `RequestPacket` struct so that multiple destinations' requests can be in the same block.

Looking at `RequestReceiver.hh`, I think `requests_` and `request_timing_` could be combined to a map between `sequence_id` and a struct containing timestamp, time received, and destination. We could then provide a set of methods for each of those three, and one for returning the struct itself.

**#9 - 05/09/2019 12:40 PM - Eric Flumerfelt**

- *Due date set to 05/09/2019*

due to changes in a related task: [#22569](#)

**#10 - 05/09/2019 02:05 PM - Eric Flumerfelt**

- *Tracker changed from Feature to Idea*

**Files**

---

20190507_165208.jpg	958 KB	05/08/2019	Eric Flumerfelt
20190507_165152.jpg	895 KB	05/08/2019	Eric Flumerfelt