ME Journal

Sean Keene (Template by Tylor Slay)

last updated: September 17, 2021

Contents

List of Errors

List of Warnings

Fall 2020

hyperref

August 20, 2021 – Reorganization and Journal Update

OBJECTIVE:

I need to establish mise en place for development of the ME. **OUTLINE:**

- Reorganize my directory, put old files in an Outdated folder.
- Restart this journal.
- Update github README.md.
- Create ModelController.py

PROCEDURE:

Should all be self explanatory.

PARAMETERS:

The point is to get ready to work without fumbling over old files or struggling to get my journal in order. The directory reorganization is pretty straight forward. Restarting this journal starts with this entry, but also organizing my journal update plan (which should be: update the journal every time you push, not commit.) I've already created ModelController.py and added the class outline to it in class form; most of the daily work from here on out will be adding classes and methods one by one, along with testing.

The README needs refactored, and some documentation should be made public. I need to decide what gets links and what, if anything, gets added to the repository directly.

OBSERVATIONS:

I did it. Needs more documentation, but I'll figure out the most useable way of handling that in the future.

DATA:

N/A

RESULTS:

I'm done enough to be ready to move on with programming next week.

Aug 21, 2021 – EdmCore and Callback Classes: step 1

OBJECTIVE:

Get started on EDMCore. See if I can get a simulation running.

OUTLINE:

- Retrieve useful code from the Log Test.
- Start filling in the methods.
- Add startup calls to the program's run routine.

PROCEDURE:

Code them.

PARAMETERS:

N/A

OBSERVATIONS:

Got it done. Everything worked smoothly. The MC is now able to initiate and run simulations from GridAPPS-D. It has the old issue where nothing stops, I'll have to hard code that in. Bigger issue: I started working with callback classes. The commands don't work. The sim runs, but nothing happens. The callbacks aren't called when expected. Need to talk to PNNL about this.

DATA:

All recorded data, or where it can be found.

RESULTS:

EDMCore is a bit more done than I was expecting, since I was able to do things like add config.txt. Callbacks are the next thing to tackle. I knew they'd be a problem, so this is fine. Consult with PNNL. Will report back in next entry.

Aug 23, 2021 – Callback Class Troubleshooting/Repair

OBJECTIVE:

TS&R Callback classes

OUTLINE:

- Contact PNNL
- Fix EdmMeasurementProcessor
- Fix EdmTimeKeeper
- Fix EdmOnClose

PROCEDURE:

Currently using the old callback functions to call the `on_message` method of each callback class. Frankly, this works perfectly well and is invisible to the user and developers in most cases. It does require function definitions and global variables and weird stuff that I'd prefer to remove. So, after contacting PNNL, figure out the issue and fix the classes.

PARAMETERS:

N/A

OBSERVATIONS:

The issue was in the way I was gathering the simulation ID. I reused code that I'd never actually USED in the old script, and this code turns out not to have been working this whole time. Turns out there's an easier way to do it anyways; I just have to start the simulation first, and then get the ID, then use that to instantiate everything. It works fine.

The EdmMeasurementProcessor works fine. The EdmTimekeeper also works fine, but differently from the old callback function: it uses a subscription to the log topic and therefore I have to filter out the messages that include timestep incrementation manually. This is done, and works more or less fine. The EdmOnClose isn't yet working since there's no obvious subscription topic for it. I may have to settle for the callback function on this one, but I'm contacting PNNL to be sure.

May be a timekeeping bug. The timestep increments before the measurement processor; but, the measurement processor uses its own timestamp anyways. Timecodes seem to be 2 off. Will need to do some rigorous testing at the end of Phase 1 to fully verify what's going on and make sure that inputs are being reflected in the outputs at the proper time (which should be the timestep after they're injected.)

DATA:

N/A

RESULTS:

EdmMeasurementProcessor and EdmTimeKeeper working as intended. EdmTimeKeeper filtering function needs cleaned up. EdmOnClose not yet implemented. Awaiting PNNL support.

Sep 2, 2021 – Phase 1 completion

OBJECTIVE:

Determine a data scheme for internal data (I.E. grid states and association data passed to the GO and logger.) Implement MCOutputLog Run a simulation and get output logs

OUTLINE:

- Select data scheme (DICT OF DICTS)
- Add processing to EDMMeasurementProcessor
- Add accessor function to EDMMeasurementProcessor
- Write MCOutputLog
- Add accessor-logger routine to timekeeper
- Add log writing routine to timekeeper
- Add file closeout function to timekeeper closeout
- Test
- Add DER-EMs to model
- Add McInputInterface with a test function
- Add DERSHistoricalDataInput with a test function
- Test (Phase 1 closeout)

PROCEDURE:

NOTE: I accidentially overwrote a bunch of information about previous steps, but it sums up to "it works". Info starts at "Add DER-EMs to model" for this sprint.

PARAMETERS:

N/A

OBSERVATIONS:

Had a bunch of issues adding DER-EMs. A lot of them came from the fact that I was jumping back and forth between CIMHub and the Powergrid-Models repository. USE POWERGRID-MODELS FOR EVERYTHING. It grabs stuff from CIMHub, but the measurement stuff is in powergrid-models as well. A batch script could automate it, if written.

Here is how the process works:

1. Create a text file containing the feeder ID and information about each DER. (See EGoT13_der.txt in the powergrid-models/platform/DER folder.)
2. Edit powergrid-models/platform/cimhubconfig.json to read "blazegraph_url": "http://localhost:8889/bigdata/sparql".T was an issue for some reason.
3. Edit powergrid-models/platform/DER/insert_der.sh with the new filename. Run it. This will create a uuid file with IDs for the new DERs.
4. Run the sparql query in the Data section below to verify DER-EMs have been added to the system.
5. Edit/run powergrid-models/platform/list_all_measurements.sh
6. Edit/run powergrid-models/platform/insert_all_measurements.sh

I added three test DERs to the model at node 671. There were some initial struggles related to controlling them, but these all turned out to be due to incorrect mRIDs and typos. For results, see /Log Demos/testlog(9-17 binary input demo).csv. The DERs were set to 0 or 1kW, counting up in binary. The results were as expected once we knew what we were looking at: the measurement points all seem to be upstream of the inverter (and thus interconnected) and represent a single phase. So on each phase at node 671, we get 333W of power consumed per DER-EM that's "turned on". When all three are turned on, we get 1kW per phase, or 3kW total. This is as expected.

Once I understood what I was looking at, the rest of the model came into focus. The only other dynamic load in the system is a house model that seems to be disconnected until the third (measurement) timestep. Those together with the DER-EMs had noticeable and consistent effects on current throughout the system. Apart from the house and its components, we have very basic energy consumers, compensators, and node to node line segment measurements, etc.

I spent a long time trying to figure out the data scheme for the test input because I didn't understand that assignment is just a reference and I kept popping the "Time" data from an individual row in my input readout. Once I figured that out, the input manager and historical data DER-S worked pretty well. In this current state, the log columns need to be the mRIDs for the battery inverter control IDs (see query below.) Once per second, that message is ssent to the input handler which automatically changes the values for each mRID to the value in the cell. A single input command is sufficient for this (and will probably be sufficient from now on).

Functional testing completed sat. There is an issue with the timecodes starting 6 seconds later than they should be, but this is minor at this point. I made a card to troubleshoot it later, it won't hold up progress.

The following Test Plans were eligible for Phase 1 Testing:

- MC04
- MC05
- DER02
- DER07
- EDM08
- EDM10

DATA:

```
# Storage - DistStorage
PREFIX r:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX c:  <http://iec.ch/TC57/CIM100#>
SELECT ?name ?bus ?ratedS ?ratedU ?ipu ?ratedE ?storedE ?state ?p ?q ?id ?fdrid (group_concat(di
 ?s r:type c:BatteryUnit.
 ?s c:IdentifiedObject.name ?name.
 ?pec c:PowerElectronicsConnection.PowerElectronicsUnit ?s.
# feeder selection options - if all commented out, query matches all feeders
#VALUES ?fdrid {"_C1C3E687-6FFD-C753-582B-632A27E28507"}  # 123 bus
VALUES ?fdrid {"_49AD8E07-3BF9-A4E2-CB8F-C3722F837B62"}  # 13 bus
#VALUES ?fdrid {"_5B816B93-7A5F-B64C-8460-47C17D6E4B0F"}  # 13 bus assets
#VALUES ?fdrid {"_4F76A5F9-271D-9EB8-5E31-AA362D86F2C3"}  # 8500 node
#VALUES ?fdrid {"_67AB291F-DCCD-31B7-B499-338206B9828F"}  # J1
#VALUES ?fdrid {"_9CE150A8-8CC5-A0F9-B67E-BBD8C79D3095"}  # R2 12.47 3
 ?pec c:Equipment.EquipmentContainer ?fdr.
 ?fdr c:IdentifiedObject.mRID ?fdrid.
 ?pec c:PowerElectronicsConnection.ratedS ?ratedS.
 ?pec c:PowerElectronicsConnection.ratedU ?ratedU.
 ?pec c:PowerElectronicsConnection.maxIFault ?ipu.
```

```
 ?s c:BatteryUnit.ratedE ?ratedE.
 ?s c:BatteryUnit.storedE ?storedE.
 ?s c:BatteryUnit.batteryState ?stateraw.
   bind(strafter(str(?stateraw),"BatteryState.") as ?state)
 ?pec c:PowerElectronicsConnection.p ?p.
 ?pec c:PowerElectronicsConnection.q ?q.
 OPTIONAL {?pecp c:PowerElectronicsConnectionPhase.PowerElectronicsConnection ?pec.
 ?pecp c:PowerElectronicsConnectionPhase.phase ?phsraw.
   bind(strafter(str(?phsraw),"SinglePhaseKind.") as ?phs) }
 bind(strafter(str(?s),"#_") as ?id).
 ?t c:Terminal.ConductingEquipment ?pec.
 ?t c:Terminal.ConnectivityNode ?cn.
 ?cn c:IdentifiedObject.name ?bus
}
GROUP by ?name ?bus ?ratedS ?ratedU ?ipu ?ratedE ?storedE ?state ?p ?q ?id ?fdrid
ORDER by ?name
```

RESULTS:

All tests completed sat. Phase 1 of the ME is now complete.