

Realtime - Task #1616

Schain v3 - modules, objects and send data realtime app

02/14/2019 06:08 PM - Luis Rodriguez

Status:	Resolved	Start date:	02/18/2019
Priority:	Normal	Due date:	02/21/2019
Assignee:	Luis Rodriguez	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	version 1.0	Spent time:	0.00 hour

Description

- Understanding of schain module and how objects related each others.
- Write/modify function to send data to app realtime(similar to client.py).
- Fix timestamp issue in realtime app (server-side)

Related issues:

Follows Task #1615: Signal chain v3	Resolved	02/11/2019	02/14/2019
Follows Task #1614: Fixed bugs on realtime app	Resolved	02/04/2019	02/07/2019
Precedes Task #1617: realtime app - schain v3	In progress	02/25/2019	02/28/2019

History

#1 - 02/14/2019 06:12 PM - Luis Rodriguez

- Follows Task #1615: Signal chain v3 added

#2 - 02/14/2019 06:12 PM - Luis Rodriguez

- Follows Task #1614: Fixed bugs on realtime app added

#3 - 02/14/2019 06:19 PM - Luis Rodriguez

- Precedes Task #1617: realtime app - schain v3 added

#4 - 02/20/2019 02:18 PM - Luis Rodriguez

- Status changed from New to In progress

- % Done changed from 0 to 40

#5 - 02/26/2019 07:48 PM - Luis Rodriguez

- % Done changed from 40 to 70

- Se utilizo el entorno de debugger de python (import pdb; pdb.set_trace()) para el analisis del script
- Se hizo la prueba con la operacion SpectraPlot se agrego el parametro plot_server = tcp://localhost:4444 para la conexion con la app de realtime
op_spectra_plot = u_proc_spectra.addOperation(name='SpectraPlot', optype='external') op_spectra_plot.addParameter(name='plot_server', value='tcp://localhost:4444', format='str')

#6 - 03/05/2019 10:38 PM - Luis Rodriguez

- Status changed from In progress to Resolved

- % Done changed from 70 to 100

En Signal Schain v3 python 3.6

* Se modifico el metodo jsonify del objeto PlotterData del modulo jrodata.py. Dicho metodo, jsonify, se encarga de estructurar el JSON adecuadamente de acuerdo a como se recibia en la app realtime web.

* Se implemento el metodo `send_to_realtime_app` que se encargaba de llamar al metodo `jsonify` para luego separar la llave 'data' y finalmente enviar cada uno, con sus respectivo metadata, a la app empleando el metodo `send_to_server`. Ambos metodos metodos, `send_to_realtime_app` y `send_to_server` pertenecen al objeto `Plot` en el modulo `jroplot_base.py`.

* En el script principal es importante agrega el codigo del experimento esto se realiza agregando un parametro a la operacion de `SpectraPlot` de la siguiente manera: `op_spectra_plot.addParameter(name='exp_code', value='191', format='str')`.

En la app realtime web:

* Se modifiko la seleccion de cada ploteo, ahora emplea una condicion que busca dentro de una lista.

* El tiempo de espera de cada cada se ajusta de acuerdo a su intervalo (`time.sleep(interval)`)

* Que pendiente el ultimo punto de esta tarea 'Fix timestamp issue in realtime app (server-side)'