FUSION
FOR
ENERGY

MARTe2 Users Meeting
HTTP

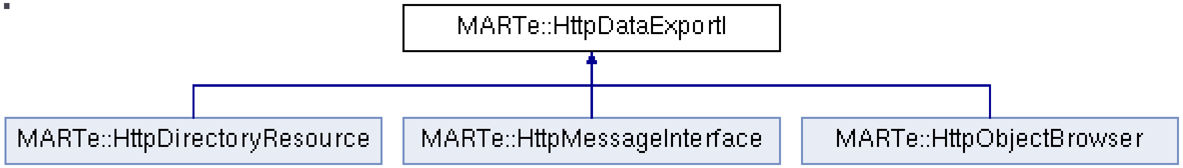
Andre Neto, Filippo Sartori

May, 2019

HTTP

Interaction with MARTe2 based applications using the HTTP protocol

- Allows to query live information (using JSON) about any Object through the **ExportData** method.



- Classes can also implement the HTTPDataExportI interface
  - **GetAsStructuredData** – StructuredDataI interface to the object
  - **GetAsText** – free text (e.g. HTML)
  - Type of request set by including in the URL query string:TextMode=0 or TextMode=1.

Warning

The data model and the data presentation layers should be as decoupled as possible.
For the majority of use-cases, objects should not implement the HTTPDataExportI interface.
Objects should expose data using the ExportData method.
It is assumed that all the data visualisation features are delegated to the HTTP client.

HttpObjectBrowser

HTTP browsing of any ReferenceContainer. Allows to query information about any component in a given MARTe application instance.

- Accessing to an HttpObjectBrowser with **TextMode=1**
  - Checks if there is an **HttpDirectoryResource** child instance capable of handling the request and if so forwards the request to it.
  - This mechanism can be used, e.g. to return an index.html page.

```
+WebRoot = {
   Class = HttpObjectBrowser
   Root = "." //Navigation w.r.t. to this object
   +ObjectBrowse = {
      Class = HttpObjectBrowser
      Root = "/" //Navigation w.r.t. to the ObjectRegistryDatabase root
   }
   +ResourcesHtml = {
      Class = HttpDirectoryResource //If TextMode=1 will try to see if the component can resolve the request
      BaseDir = "../../../../../Resources/HTTP/"
   }
   …
```

HttpMessageInterface

Send commands to MARTe applications using the URL query string. Commands are interpreted and forward as MARTe messages.

```
...
+HttpMessageInterface = {
    Class = HttpMessageInterface
    +GOTOIDLE = { //Trigger the message GOTOIDLE if the URL contains msg=GOTOIDLE
        Class = Message
        Destination = StateMachine
        Function = GOTOIDLE
        Mode = ExpectsReply
    }
...
```

JavaScript client

Simple JavaScript based client framework that aims at easing the development of graphical user-interfaces.

- Plug-in mechanism that retrieves data from the server (using JSON) and offers it to a given class that inherits from the MARTeObject JavaScript class.

```
...
class HttpPendulumEx1 extends MARTeObject {
...
    displayData(jsonData) {
        var angle = parseFloat(jsonData["angle"]);
        var width = this.canvas.clientWidth;
        var height = this.canvas.clientHeight;
...
```

Note

For more complex client applications, users should use the JSON interface and design their own client applications.

MARTeLoader

Allows to allocate a given instance of a user provided JS class to an HTML element and later associated to a JSON data source.

```
...
var mainTargetContainer = document.getElementById("mainTarget");
//Associate the HttpPendulumEx1P2.js to an HTML element
//identified by the id "table0x1" and load with data coming from http://.../Pendulum
MARTeLoader.instance().load("Pendulum", "HttpPendulumEx1P2", "table0x1");
...
```

url

js class name

HTML id

Exercises

- ExportData
- HTTPDataExportI
- OPI

HttpObjectBrowser

Objective: navigate the HttpObjectBrowser

```
cd ~/Projects/MARTe2-demos-padova/Startup/
./Main.sh -l RealTimeLoader -f ../Configurations/RTApp-HTTP-1.cfg -m
StateMachine:START
```

- Open a browser and point at https://127.0.0.1:8084
- Navigate the tree to find ObjectBrowse->TestApp->Functions->FixedGAM1

Success: application executes and can be introspected using the browser

ExportData

Objective: change the data that is shown the browser

- Modify GAMs/FixedGAMExample1/FixedGAMExample1.cpp
- Change the ExportData to export the **offset** variable
- Compile

```
cd ~/Projects/MARTe2-demos-padova/
export MARTe2_DIR=~/Projects/MARTe2-dev
export MARTe2_Components_DIR=~/Projects/MARTe2-components/
make -f Makefile.x86-linux
```

- Run the application

```
cd ~/Projects/MARTe2-demos-padova/Startup/
./Main.sh -l RealTimeLoader -f ../Configurations/RTApp-HTTP-1.cfg -m
StateMachine:START
```

- Open a browser and point at https://127.0.0.1:8084
- Navigate the tree to find ObjectBrowse->TestApp->Functions->FixedGAM1

Success: application executes and can be introspected using the browser and the offset variable is shown.

Plug-ins

Objective: change the javascript that is used to display data of a component

- Modify Resources/HTTP/FixedGAMExample1.js
- Change the **displayData** to show only the “InputSignals”
- Run the application

```
cd ~/Projects/MARTe2-demos-padova/Startup/
./Main.sh -l RealTimeLoader -f ../Configurations/RTApp-HTTP-1.cfg -m
StateMachine:START
```

- Open a browser and point at https://127.0.0.1:8084
- Navigate the tree to find ObjectBrowse->TestApp->Functions->FixedGAM1

Success: application executes and can be introspected using the browser to show only the InputSignals.

OPIs

Objective: create a simple HTML page that interfaces with MARTe2

- Modify Resources/HTTP/example1.html
- Change the **loadPanels** to add a new instance of **FixedGAM1** in **table1x1** with color “blue”.
- Run the application

```
cd ~/Projects/MARTe2-demos-padova/Startup/
./Main.sh -l RealTimeLoader -f ../Configurations/RTApp-HTTP-1.cfg -m
StateMachine:START
```

- Open a browser and point at https://127.0.0.1:8084/?path=example1.html

Success: application executes and the page is rendered with 4 widget instances

FUSION
FOR
ENERGY

Thank you for your attention

Follow us on:

www.f4e.europa.eu

www.twitter.com/fusionforenergy

www.youtube.com/fusionforenergy

www.linkedin.com/company/fusion-for-energy

www.flickr.com/photos/fusionforenergy