

IEEE 1451 HTTP SERVER IMPLEMENTATION FOR MARINE DATA

Joaquín del Río¹, Marc Martínez¹, Daniel Mihai Toma¹, Antoni Mànuel¹, Helena G. Ramos²
1 SARTI Research Group. Universitat Politècnica de Catalunya. Rambla Exposició 24, 08800, Vilanova i la Geltrú. Barcelona. Spain.
+ (34) 938 967 200. Joaquin.del.rio@upc.edu
2 Instituto de Telecomunicações, DEEC. Instituto Superior Tecnico, UTL. Lisboa, Portugal.

Abstract- Accessing real time data from a marine sensor network (MSN) can be a challenge. Open access to real time data using interoperable internet technologies is one of the major demands. The IEEE 1451 Smart Transducer Interface Standards [1,2] specify a standard API (application processor interface) process to discover and access sensor data by using an HTTP connection. This paper presents an HTTP Server for Marine Data coming from OBSEA[3,5] cabled Observatory at the Western Mediterranean Sea. The implementation was using LabVIEW Web Services.

Keywords- IEEE1451, LabVIEW Web Services, Marine Sensor Network

INTRODUCTION

Data Interoperability is one of the major issues nowadays, where internet technologies enable the access to sensors data in real time. Sensor Web Enable (SWE) [4] from OGC (Open Geospatial Consortium) and IEEE1451 Standard are two approaches to achieve these interoperability problems. Discover and accessing available sensor data from an Internet web server through a standard is one objective of this work. In this case, an HTTP 1451.0 Server has been implemented using LabVIEW Web Services. Previous implementations of a HTTP 1451.0 server were done using different technologies or language programming like JAVA. In this case, LabVIEW graphical programming was used to facilitate the integration process for electronics or scientist who most times are more familiar with LabVIEW than other languages.

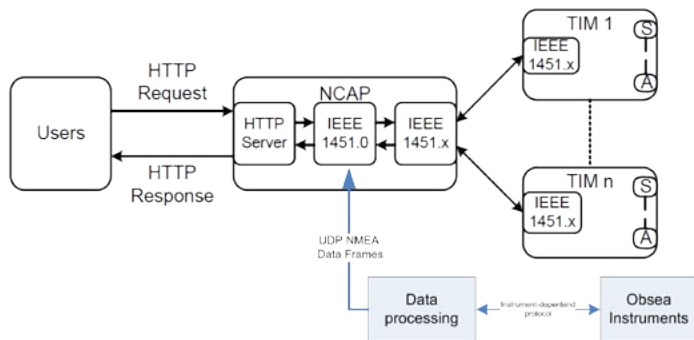


Figure 1. HTTP access to an IEEE 1451.0 NCAP and Non Standard Instruments Interface

IMPLEMENTATION

The IEEE 1451 family of standards were created to address the standard digital documenting of sensors [1,2]. The main objective is to add a digital layer of memory, functionality and communication to sensors so as to make sensors controllable and measurements accessible through a network with sufficient information on the sensor characteristics and history.

Figure 1 depicts the main components of a full compliance IEEE1451 system: The Transducer Interface Module (TIM), a device that could be either a sensor or an actuator with transducer channels and a Network Capable Applications Processors (NCAP) where different instruments or transducers are connected through a TIM (Transducer Interface Module). Each NCAP inside the system will have a NCAP address and each TIM connected to NCAP his TIM addresses.

For a non IEEE1451 compliance instruments, a data processing block implement the instrument manufacture-dependent protocol and translate the instrument data to a UDP NMEA-0183 style data frame that is sent to the IEEE1451.0 NCAP who interpret this data coming from a virtual TIM with virtual channels.

As an example, a non IEEE1451 instrument likes a marine CTD who measures current, temperature and depth will be interpreted by our NCAP as a virtual TIM with three different channels.

Some of the API commands implemented by the LabVIEW Web Service are shown in Table 1

A remote client use the "http" protocol to send messages to the IEEE 1451.0 NCAP, which acts like a server, serving transducer data to the remote client. The HTTP message from the remote client unit will be transmitted across a network connecting the remote client with the IEEE 1451.X transducer nodes, or in this case to the non IEEE1451 instruments. The message will adhere to the HTTP URL syntax (RFC 2616) as `http://<host>:<port>/<path>?<parameters>`.

Table 1. Implemented Commands by the LabVIEW HTTP 1451 web Service

•TimDiscovery (TimDiscoveryHTTPRequest, TimDiscoveryHTTPResponse)
•TransducerDiscovery (TransducerDiscoveryHTTPRequest, TransducerDiscovery-HTTPResponse)
•ReadTransducerData (ReadTransducerDataHTTPRequest, ReadTransducerDataHTTPResponse)
•ReadMetalDTEDS (ReadMetalDTEDSHTTRequest, ReadMetalDTEDSHTTResponse)
•ReadTIMGeoLocationTEDS(ReadTIMGeoLocationTEDSHTTRequest, ReadTIMGeoLocationTEDSHTTResponse)

For example: <http://esonet.epsevg.upc.es:1451/1451/Discovery/TIMDiscovery?ncapId=1&responseFormat=xml>

Where the NCAP server is hosted at esonet.epsevg.upc.edu at port 1451. The LabVIEW web service path for the discovery messages is /1451/Discovery and in this example a TIMDiscovery request to our NCAP is done asking for an xml formatted reply.

The LabVIEW IEEE1451 HTTP web service reply to the previous command will be `<TIMDiscoveryHTTPResponse xsi:schemaLocation="http://localhost/1451HTTAPI http://grouper.ieee.org/groups/1451/0/1451HTTAPI/TIMDiscoveryHTTPResponse.xsd"><errorCode>0</errorCode><ncapId>1</ncapId><timIds>1,2</timIds></TIMDiscoveryHTTPResponse>`

Where two TIMs with TimIds 1 and 2 are present in our system.

Our system is composed by one NCAP server implemented in LabVIEW as a 1451 Web Service and for the first test two TIMs who represents one CTD instrument with three and a meteorological Data Station with 40 channels

- CTD Instrument TIMid=1. Three Channels: conductivity, temperature and depth.

- Meteo Station TIMid=2. Forty Channels: *stationDate, stationTime, outsideTemp, wind10Avg, windDir, barometer, outsideHumidity, solarRad_uv, hiOutsideTemp, lowOutsideTemp, hiHumidity, lowHumidity, hiBarometer, lowBarometer, hiWindSpeed, hiSolarRad, hiUV, dailyRain, wind10Avg, hiMonthlyOutsideTemp, lowMonthlyOutsideTemp, hiMonthlyHumidity, lowMonthlyHumidity, hiMonthlyBarometer, lowMonthlyBarometer, hiMonthlyWindSpeed, hiMonthlySolarRad, hiMonthlyUV, monthlyRain, hiYearlyOutsideTemp, lowYearlyOutsideTemp, hiYearlyHumidity, lowYearlyHumidity, hiYearlyBarometer, lowYearlyBarometer, hiYearlyWindSpeed, hiYearlySolarRad, hiYearlyUV, totalRain*

CONCLUSION

In order to facilitate the interoperable access to the Obsea instruments data, a LabVIEW IEEE1451 HTTP web server has been developed. This development allow the OBSEA technicians to share instrument data through a standard interface easily and quickly.

REFERENCES

- [1] L. Kang, "IEEE 1451: A standard in support of smart transducer networking," presented at Instrumentation and Measurement Technology Conference, 2000. IMTC 2000. Proceedings of the 17th IEEE, 2000.
- [2] E. Y. Song and L. Kang, "Understanding IEEE 1451-Networked smart transducer interface standard - What is a smart transducer?," Instrumentation & Measurement Magazine, IEEE, vol. 11, pp. 11-17, 2008.
- [3] Mànuel, A.; Noguerras, M.; Del Río, J. "OBSEA: an expandable seafloor observatory" *Sea technology*, ISSN: 0093-3651
- [4] M. Botts, G. Percivall, C. Reed, and J. Davidson, "OGC Sensor Web Enablement: Overview and High Level Architecture."
- [5] Aguzzi, Jacopo; Mànuel, Antoni; Condal, Fernando; Guillén, Jorge; Noguerras, Marc; del Río, Joaquin; Costa, Corrado; Menesatti, Paolo; Puig, Pere; Sardà, Francesc; Toma, Daniel; Palanques, Albert. 2011. "The New Seafloor Observatory (OBSEA) for Remote and Long-Term Coastal Ecosystem Monitoring." *Sensors* 11, no. 6: 5850-5872.