EPJ Web of Conferences **253**, 04008 (2021) https://doi.org/10.1051/epjconf/202125304008
ANIMMA 2021

CAREDAS: a Comprehensive Architecture for a Redundant and Evolutive Data Acquisition System for JHR reactor

Fabrice Leroux[1], Lionel Ducobu[1] and Frédéric Milleville[1]

[1]CEA Cadarache, DSTG/STIC, France
Corresponding author: fabrice.leroux@cea.fr

***Abstract*—A new material testing reactor Jules Horowitz Reactor is under construction at CEA Cadarache. The materials to be irradiated will be placed into experimental devices around the reactor. Process and measurements of experimental devices will be carried out by command control. A data acquisition system having processing performances will be associated to the programmable logic controller. The challenge is to design and realize for twenty experiment devices a high availability data acquisition system architecture for 50 years of sustainability. The real time target will achieve 24/7 data acquisition and real time processing. This scalable architecture could be use as well for JHR experimental devices with high availability as for testbed. This architecture could be run on a standalone station for a measuring bench or deployed on cluster for high availability. CAREDAS's design is modular and use proven widely used open source solutions. All parts are independent from each other and can be replaced with another technology with the same functionalities. This ensures sustainability and control of software sources.**

***Keywords* —CAREDAS, MQTT, InfluxDB, Grafana, data acquisition system, DAQ, ExprTk, TSN, EtherCAT, OPC UA.**

I. INTRODUCTION

A new material testing reactor Jules Horowitz Reactor (JHR) [1] is under construction at CEA Cadarache. The materials to be irradiated will be placed into experimental devices around the reactor. Process and measurements of experimental devices will be carried out by command control (Fig. 1).

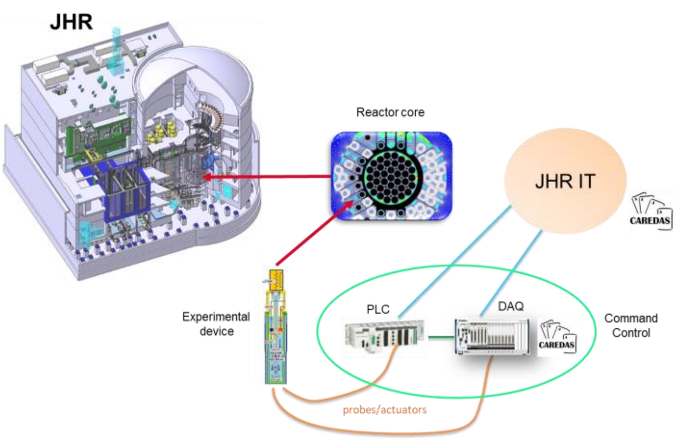

Fig. 1. Control command of JHR experimental devices.

Programmable Logic Controller (PLC) dedicated on each experimental device will realize this function. Some experiment devices will need to achieve complex real time processing that PLC cannot reach. A data acquisition system (DAQ) having these processing performances will be associated to the PLC. This system will realize measurements and processing. The challenge is to design and realize for twenty experiment devices a high availability data acquisition system architecture for 50 years of sustainability. The real time target will achieve 24/7 data acquisition and real time processing. The other components will be in charge of data storage, online and offline data visualizations, experiment setting and processing without modifying software core.

This scalable architecture could be used as well for JHR experimental devices with high availability as for testbed. This architecture could be run on a standalone station for a measuring bench or deployed on cluster with redundant data acquisition system for high availability. CAREDAS's design is modular and use proven widely used open source solutions of IIoT world (Industrial Internet of Things). All parts are independent from each other and can be replaced with another technology with the same functionalities. This ensures sustainability and control of software sources.

The real-time DAQ is modular according to the user's needs with various inputs/outputs such as analog inputs, digital inputs, temperature sensor. A DAQ management software, which can be used by a non-IT specialist, combines the functions of parameters setting, supervision and visualization of measurements. Real-time processing is defined in this software with a scientific programming language such as Scilab.

This architecture is modular and scalable according to the needs. It offers high availability on demand at all levels.

II. CAREDAS ARCHITECTURE

CAREDAS architecture (Fig. 2) was designed to ensure high availability for 20 independent acquisition systems and adaptable according to the needs of the plant. It ensures the archiving of a large volume of data during experiment. The acquisition chain has a sustainability of at least 50 years. Solutions selected are proven and if possible open source. All the functions are synchronized by Network Time Protocol (NTP) that distributes a clock. DAQ are synchronized by Time Sensitive Network (TSN).

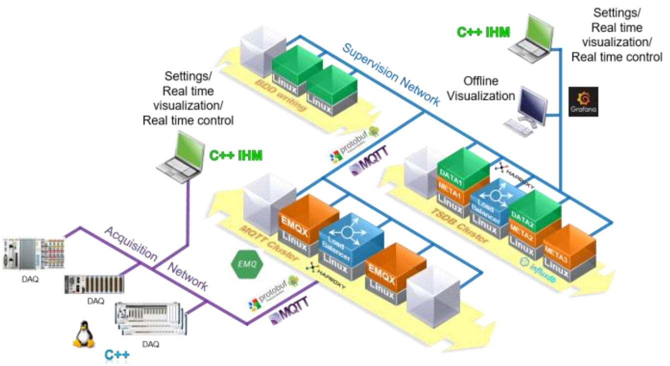

Fig. 2. CAREDAS architecture.

At the bottom of the chain, there are real-time DAQ allowing continuous acquisition of up to 300 sensors while performing 200 real-time processing for 50ms cycle time. No data loss is provided by redundant DAQ. DAQ software is independent of the hardware on which it runs. Linux operating system with real-time functionality is used.

The user can, through a configuration tool, configure his experience, add or remove measurement channels, hardware without software programming. He will define his real-time processing in a structured scientific programming language without programming DAQ. A DAQ supervision and a real-time visualization of measurements allow to check settings as close as possible to the hardware.

At the top of the chain, data and system settings archiving ensure high availability of storage and access to data. The database selected is InfluxDB [2], a Time series database (TSDB) which is able to absorb and store a large amount of data. For example, a DAQ with the maximum number of channels and 50ms of cycle time represents an amount of 14 GB/day.

The real-time acquisition chain is made up of two networks: the supervision network and the acquisition network. The supervision network and the acquisition network are common to all DAQ and are based on separate Ethernet networks. This division isolates and secures DAQ. The acquisition network must ensure no data loss and its isolation allows to not be disturbed by traffic that the connected systems are not receiver.

Communication between all these functions is ensured by messaging using the open source MQTT (Message Queuing Telemetry Transport) protocol [3] that is widely used and proven.

Offline data visualization is achieved with Grafana to compose dashboards that query directly InfluxDB databases.

Only three software in C++ language are homemade to ensure sustainability: real time data acquisition, writing data into InfluxDB, tools for experiment setting and supervision.

All software run on station or cluster nodes on the Linux operating system (Red Hat distribution).

III. COMMUNICATION BETWEEN FUNCTIONS: MQTT

MQTT messaging is the heart of CAREDAS. It manages the dialogue between the different functions of the acquisition chain without loss of messages. MQTT is an open source messaging protocol widely used in the world of Internet of Things (IoT) and in Web as in monitoring servers of data center for example. MQTT is based on publish/subscribe principle. A server named "broker" is responsible for collecting messages published by clients and distributes them to subscribed clients. A client has the possibility to define a level of quality of service for message transmitted. The level of quality tell on broker if it must or not ensure that the message have been received by clients.

In view of the number of measurements transmitted by DAQ in each cycle, it is not reasonable to transmit one MQTT message per measurement. Measurements are grouped to optimize message transmission. A data serialization, Google Protocol buffers [4], is using to make a MQTT message (Fig. 3).

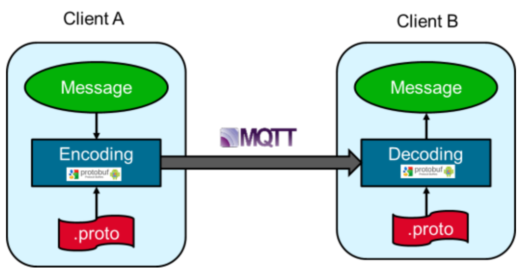

Fig. 3. Google Protocol buffers.

It was developed by Google and available under an Open Source License. Protocol buffers is a language for describing a data structure that allows a program to produce or read a message.

MQTT CAREDAS architecture (Fig. 4) is composed of a cluster of three brokers and two load balancer. The cluster ensures high availability of MQTT messaging. In case of a node is lost in the cluster or a network failure, other nodes take over. The selected open source MQTT broker is EMQ X from EMQ [5].

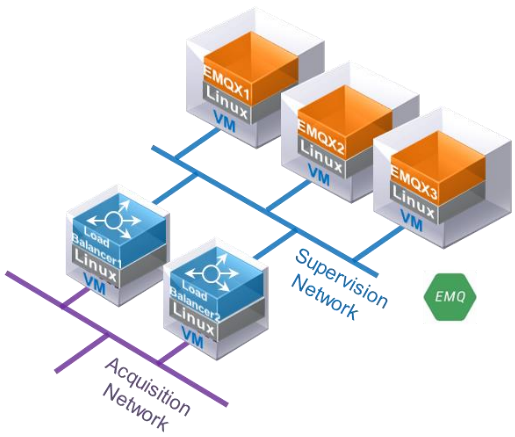

Fig. 4. CAREDAS MQTT architecture.

The load balancer is the single point of entry between clients and brokers. It distributes the connection requests from clients to a broker with the least load. Brokers are only connected with the load balancer. It separates the supervision network from the acquisition network.

https://doi.org/10.1051/epjconf/202125304008

IV. Data Archiving

InfluxDB is the database to store DAQ measurements. Data storage is based on a cluster architecture that is a collection of nodes or servers (Fig. 5). A cluster enables high availability and load balancing across multiple nodes. Like MQTT cluster, a load balancer distributes requests from clients to InfluxDB nodes following nodes load. Data are automatically replicated on all cluster nodes.

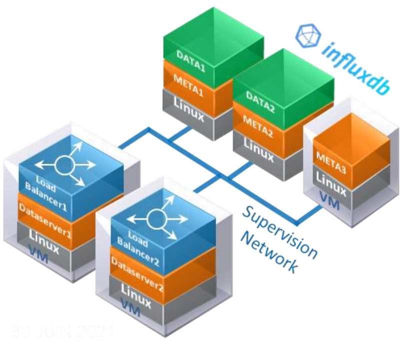

Fig. 5. CAREDAS data archiving.

The software in charge of writing data into InfluxDB database is homemade. It receives data from DAQ by MQTT message and write it in a dedicated DAQ database. This software makes the arbitration and consistency test of data received from redundant DAQ before storing it.

V. Data Visualization

Two types of data visualization are possible: online and offline.

Online data visualization displays data directly from DAQ in subscribing to MQTT messages (Fig. 6). It is reachable in the configuration tool. The user can tune DAQ parameters and displays the result with the same tools near the DAQ.

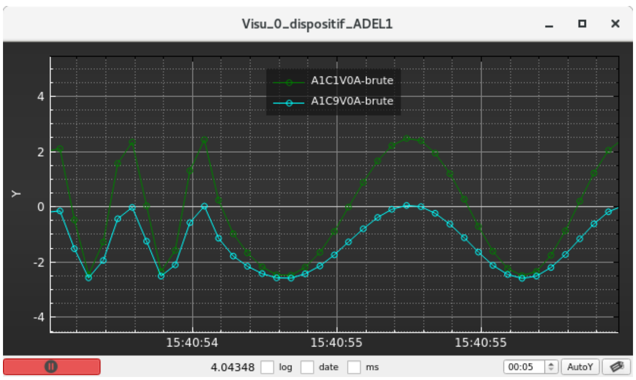

Fig. 6. Online data visualization.

Offline visualization displays data from InfluxDB databases with Grafana [6]. Grafana is an open source software to display data into dashboards. User makes dashboard (Fig. 7) with various panel like graph display, gauge, table of values. Dashboard are designed and reachable in a web browser. Alarms on threshold or data transformation can be performed in dashboard.

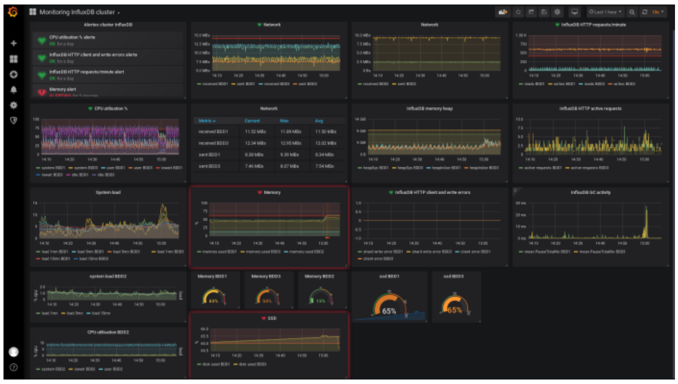

Fig. 7. Grafana dashboard.

A Grafana plugin allows to display diagrams on dashboard with objects animated by data value like SCADA for industrial process survey (Fig.8). Objects appearance can be change on data threshold like color change in a rule.

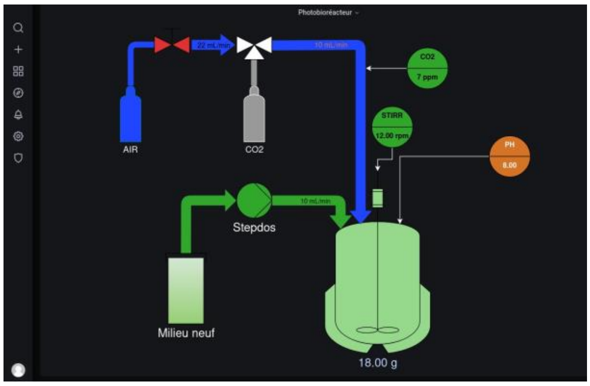

Fig. 8. SCADA on Grafana dashboard.

Grafana run as service on a Linux server. A high availability is getting by running many Grafana services on different nodes.

VI. Data Acquisition System

Real time data acquisition software is modular and technology independent (Fig. 9). It can compose of many chassis with different technologies following the user needs. Proven industrial system technologies have been selected to ensure sustainability on 10 years. Technologies implemented are PXIe bus, CompactRIO and CompactDaq from National Instruments, EtherCAT protocol and various measurement units over Ethernet. The operating system is Linux with PREEMPT_RT real time functionality. CAREDAS data acquisition software can also run on a Raspberry PI. To ensure the high availability and no data loss many mechanisms are implemented. First, data are stored locally on a disk. MQTT client of the DAQ stores messages in memory while few minutes during a network loss. Messages stored in memory will be transmitted if the link is restored before this period. If this deadline is exceeded then data stored on disk will be transmitted. DAQ can also be redundant with three systems.

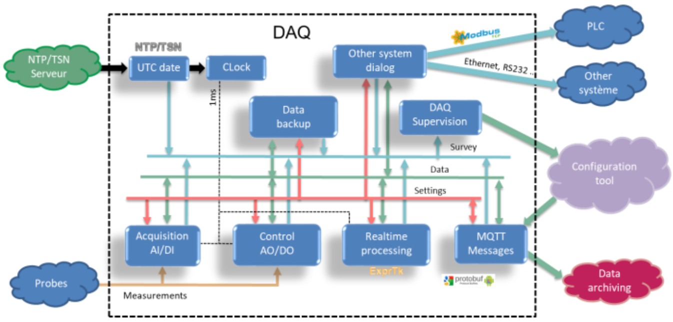

Fig. 9. DAQ architecture.

DAQ provides data timestamp in UTC (Universal Time Coordinated) with a 1ms, 1µs or 1ns precision. Electronic board synchronization is realized by distributed hardware clocks. These clocks are generated from system clock that is synchronized by TSN protocol [7]. Thus, all DAQ are synchronized at 1µs.

DAQ software is homemade in C++ language. It is modular, technology, operating system and hardware independent. This architecture allows adding new hardware or replace a library without impact on all DAQ software.

Real time processing are describing in configuration tool, thus user do not need to modify DAQ software. An open source mathematical expression parser ExprTk [7] performs real time processing. On the start of data acquisition, ExprTk produces an instance with an AST (Abstract Syntax Tree) of the real time processing and use it in run time to evaluate ExprTk expression. The ExprTk library has capabilities of mathematical functions, operators, control and loop structure and user function that permit to realize complex processing like digital processing or machine learning.

VII. Data Acquisition Setting

The configuration tool (Fig. 10) allows user to define intuitively all necessary information for DAQ. It surveys the DAQ operation and visualizes online data in real time directly from DAQ MQTT messages. The user can tune DAQ parameters near the DAQ with a laptop.

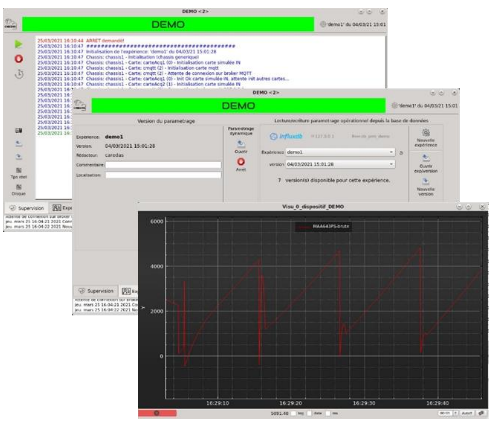

Fig. 10. CAREDAS configuration tool.

He can define an unlimited number of configuration stored in InfluxDB database. Every parameter backup generates a new configuration version. The tool is able to generate the CAREDAS configuration for all the acquisition chain when adding a new DAQ (MQTT topics, create database, DAQ system configuration). The configuration tool is a homemade software in C++ language usable on any operating system.

Multiple tabs allow to simply define chassis, electronic boards and channels. Hardware channels are defined in channel tab and associates the hardware channel to the measurement name in database. Real time processing are written in this tab with the possibility to create user functions in a library and share variables between real time processing or user functions (Fig. 11).

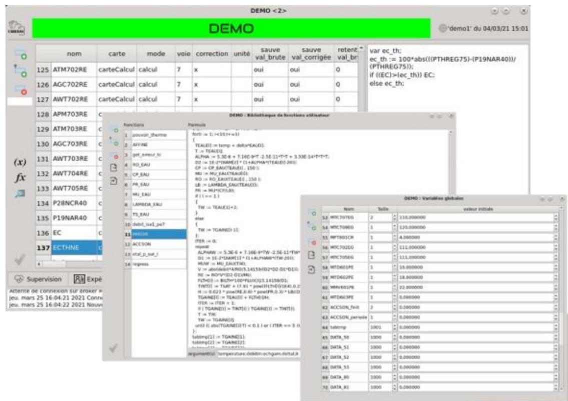

Fig. 11. Real time processing in configuration tool.

VIII. CAREDAS as Single Station

All the CAREDAS architecture is able to run on single station like personal computer (PC) without software modification (Fig. 12). On this PC run EMQ X MQTT broker, InfluxDB database, offline data visualization with Grafana and CAREDAS configuration tool. CAREDAS data acquisition software can also run on this station when the hardware is not open like a black box. CAREDAS is then adapted to a standalone measuring bench.

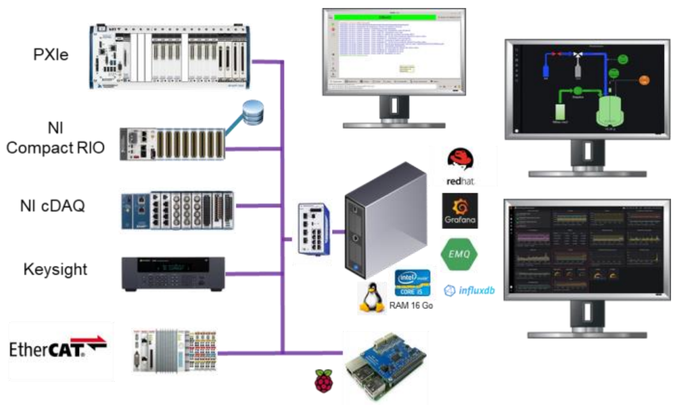

Fig. 12. CAREDAS single station architecture.

IX. Conclusions

A proof of concept has been realized by the end of 2019. Data flow of 20 DAQ experimental devices were simulated 24/7 for two months without data loss (Fig. 13). Experimental devices had 800 channels each for a cycle time of 50ms. Two real DAQ with CompactRIO and PXIe DAQ are used and Raspberry PI 3 simulates the 18 others. This represents 16000 measurement channels every 50ms ether a flow of 320000 measurements by second and 28 billion measurements by day. MQTT cluster is made of three DELL R630 dual processor server with 64 GB of RAM to store messages in case of client network failure. InfluxDB cluster is composed of two data nodes and three meta nodes with SSD disk for databases. Computers are DELL R630 dual processor server with 16 GB of RAM.

Fig. 13. DAQ experimental devices for CAREDAS POC

Now, CAREDAS is under development and a first stable version for single station is planning for September 2021. EtherCAT [9] and OPC UA [10] protocol are being integrated into CAREDAS to have a wide choice of technologies. Meanwhile, a beta version was deployed on five measuring bench on some CEA Cadarache plant. This strategy allowed us to have feedback of CAREDAS use and improves software along developments. CAREDAS version for JHR is planned for March 2022.

References

[1] JHR Web site http://www-rjh.cea.fr
[2] Influx DB from influxdata https://www.influxdata.com/
[3] MQTT https://mqtt.org/
[4] Google Protocol Buffers https://developers.google.com/protocol-buffers/
[5] EMQ X from EMQ https://www.emqx.io/
[6] Grafana https://grafana.com/
[7] TSN https://en.wikipedia.org/wiki/Time-Sensitive_Networking
[8] ExprTk http://www.partow.net/programming/exprtk/
[9] EtherCAT https://www.ethercat.org/
[10] OPC UA https://opcfoundation.org/about/opc-technologies/opc-ua/