

Proceedings

# Open Source Monitoring System for IT Infrastructures Incorporating IoT-Based Sensors <sup>†</sup>

Alejandro Mosteiro Vázquez \*<sup>id</sup>, Carlos Dafonte<sup>id</sup> and Ángel Gómez<sup>id</sup>

Centro de Investigación en TIC (CITIC)—Department of Computer Science and IT, Universidade da Coruña, Campus de Elviña s/n, 15071 A Coruña, Spain; carlos.dafonte@udc.es (C.D.); angel.gomez@udc.es (Á.G.)

\* Correspondence: a.mosteiro@udc.es

<sup>†</sup> Presented at the III XoveTIC Congress, A Coruña, Spain, 8–9 October 2020.

Published: 3 September 2020

**Abstract:** This paper introduces the development of a data center monitoring system based on IoT technologies. The system is meant to work as an administrative tool for system administrators in any environment, but mainly focused on data centers, since it integrates sensor and server status data. We are developing a system that gives a broad view of a data center, integrating server data such as CPU and memory usage or network bandwidth with room health parameters such as temperature, humidity, and power consumption or the presence sensors that indicate if there were people inside the room at the time a certain event occurred. As this is a work in progress, in this paper, we present the state-of-the-art of this subject, as well as what we expect to obtain from this project.

**Keywords:** monitoring; IoT; data center; open source; open hardware

## 1. Introduction

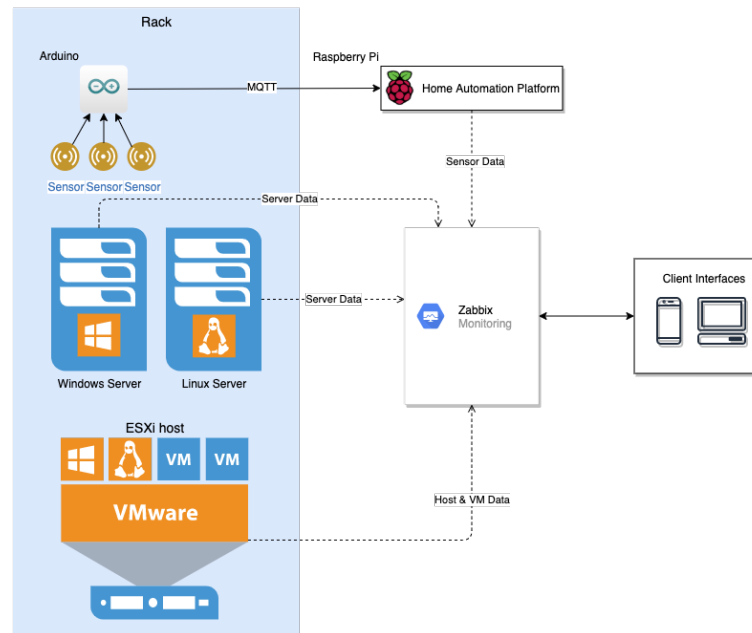
CITIC Research Centre is equipped with a modern DC in a hot-aisle disposition with ten 42U standard racks. All the structural elements in the DC—racks, cooling units, UPS—are from APC. Temperature and humidity sensors are distributed among all the racks, and data from these sensors are collected by a central server capable of sending alerts in case some values deviate from established thresholds. This solution also collects data from the UPS and cooling units, displaying real-time information about load, power consumption, heat dissipation, and cooling demand, amongst others. All of this information is centralized in their proprietary software, StruxuWare Central, and a license is needed for each node in the monitoring infrastructure. With the work described in this paper, we want to create an alternative to the current system using open source and open hardware elements and test if the final product can be as reliable as APC's proprietary system currently used in CITIC's DC and if this new solution can be easily adopted by any DC.

There are several open source monitoring platforms available for IT infrastructure, some of the most used being Zabbix, Nagios, or Icinga, as stated by several review sources such as DNSstuff [1], Opensource.com [2], or iTT Systems [3]. These platforms support similar features: network protocols, operating systems, SNMP. The main difference between them is that the former offers server performance monitoring besides network monitoring out of the box. Adding that Zabbix has a big development community, we decided to use it in this project.

As seen in Figure 1, we propose the installation of Arduino-compatible sensors into racks sending data to a Raspberry Pi with an instance of a home automation platform, such as HomeAssistant or openHAB. Then, we will design a script to communicate all gathered data to the Raspberry Pi from Arduinos to a Zabbix server. This Zabbix server is already deployed and is currently collecting data from a VMware infrastructure, its virtual machines, and several Windows and Linux servers. This way, we will have all relevant data of our DC centralized in our Zabbix server. A web client with responsive

and mobile-oriented interfaces will be developed, and it will offer users the possibility to configure alarms, thresholds, notifications, and even interact with some of the elements in the infrastructure.

**Global Architecture Diagram**



**Figure 1.** Global Architecture Diagram showing how all components communicate with each other.

We will be using DHT22 sensors for temperature/humidity monitoring and PIR sensors for presence control, all connected to Arduino-compatible boards. Specifically, we thought about using NodeMCU boards as these integrate an ESP8266 chip and allow an easy programming interface. We will also study the possibility of running these boards on battery power and test the battery life expectancy so we can install our sensors even in places where it is impossible to use the main power supplies.

There are some approaches to monitoring sensors with Zabbix like EmonTH sensors Template [4] or the ESP sensors integration for Zabbix [5]. However, these are not a complete generic solution, but templates to monitor specific sensors. Therefore, we will develop our own Zabbix monitoring templates for the chosen hardware keeping in mind that they have to be as generic as possible to monitor all kinds of sensors.

All of the above results in the two main objectives of this project: creating a centralized monitoring tool for CITIC’s IT staff and verifying that a system composed of open source and open hardware elements can be as reliable as proprietary alternatives in a real-life working environments.

**2. Materials and Methods**

One of the main tools we are using in this project is CITIC’s virtualization infrastructure. This includes several hosts with VMware’s hypervisor, ESXi, managed by VMware’s centralization tool, vCenter, where most of the Centre’s servers are virtually hosted. All of these servers are physically located in APC racks where sensors are going to be placed. We are also using, as mentioned before, Arduino-compatible sensors, Arduino-based boards and Raspberry Pi boards.

Once our monitoring system is complete, we will collect data from both alternatives: APC StruxuWare Central and CITIC’s new monitoring system. Data will be compared, assuming the data from StruxuWare Central are correct and can be established as benchmark data, computing deviations in measurements from our open hardware sensors. If measurements from our system are in acceptable ranges, we will conclude that our system is reliable.

### 3. Conclusions

In the previous section, we described our monitoring system as an open source monitoring tool that integrates room monitoring with server monitoring using open hardware elements, and there are some articles describing similar approaches such as “Designing an open source maintenance-free Environmental Monitoring Application for Wireless Sensor Networks” [6] or “Environmental monitoring system based on an Open Source Platform and the Internet of Things for a building energy retrofit” [7]. While they offer interesting results for such monitoring applications, these articles cover different types of monitoring. The first one does not include IoT or systems monitoring, and the second one, while more focused on environmental monitoring, is oriented toward building health.

We have experience on the topic of system monitoring, with some environmental sensors on Arduino with Raspberry Pi monitoring tools or DC monitoring on Raspberry Pi with several attached sensors. Therefore, with this project, we will focus on bringing together all knowledge about the topic, integrating data from different sources into one main visualization interface with enhanced features such as real-time alerts or remote control of servers and DC cooling and power elements, offering a novel monitoring system compatible with any DC configuration by applying minor changes to our original design.

**Conflicts of Interest:** The authors declare no conflict of interest.

### Abbreviations

The following abbreviations are used in this manuscript:

IoT Internet of Things  
DC Data Center

### References

1. DNSstuff Review on Open Source Monitoring Tools. Available online: <https://www.dnsstuff.com/open-source-network-monitoring-tools> (accessed on 30 July 2020).
2. opensource.com Review on Open Source Monitoring Tools. Available online: <https://opensource.com/article/19/2/network-monitoring-tools> (accessed on 30 July 2020).
3. iTT Systems Review on Open Source Monitoring Tools. Available online: <https://www.ittsystems.com/best-open-source-network-monitoring-tools/> (accessed on 30 July 2020).
4. EmonTH Sensors Zabbix Template. Available online: <https://share.zabbix.com/scada-iot-energy-home-automation-industrial-monitoring/monitoring-equipment/emonth-sensors> (accessed on 30 July 2020).
5. ESP Sensors Integration in Zabbix. Available online: <https://github.com/letscontrolit/ESPEasy> (accessed on 30 July 2020).
6. Delamo, M.; Felici-Castell, S.; Pérez-Solano, J.J.; Foster, A. Designing an open source maintenance-free Environmental Monitoring Application for Wireless Sensor Networks. Designing an open source maintenance-free Environmental Monitoring Application for Wireless Sensor Networks. *J. Syst. Softw.* **2015**, *103*, 238–247. doi:10.1016/j.jss.2015.02.013.
7. Martín-Garín, A.; Millán-García, J.A.; Bañri, A.; Millán-Medel, J.; Sala-Lizarraga, J.M. Environmental monitoring system based on an Open Source Platform and the Internet of Things for a building energy retrofit. *Autom. Constr.* **2018**, *87*, 201–214. doi:10.1016/j.autcon.2017.12.017.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).