Data Center Environment Monitoring System

Diogo Santos
Instituto Universitário de Lisboa
(ISCTE-IUL),
Lisboa 1649-026, Portugal
Diogo_Alexandre_Santos@iscte-iul.pt

Bruno Mataloto
Instituto Universitário de Lisboa
(ISCTE-IUL),
Lisboa 1649-026, Portugal
Bruno_Mataloto@iscte-iul.pt

Joao C. Ferreira
ISTAR-IUL, Instituto Universitário de
Lisboa (ISCTE-IUL and INOV INESC
Inovação - – Instituto de Novas
Tecnologias, Lisbon Portugal
jcafa@iscte-iul.pt

ABSTRACT

The Internet of things (IoT) is applied to many cases in the smart cities topic. We apply an IoT-developed platform using LoRa communication to a Data Center to understand temperature behavior within a concentration of servers and the working behavior of these server machines. We describe our work as an IoT platform to measure temperature, humidity, and energy consumption in these data centers. In the end, the gradient temperature was found in the rack, and the increasing temperature is correlated with energy consumption and the backup routines in the night. Our developed approach can be used to understand CPU usage and related temperature and the energy consumption

CCS Concepts

• Computer systems organization → Real-time systems
• Information systems → Data centers

Keywords

IoT; Sensors; Data Center; LoRa; Data Analytics; Temperature; Energy Consumption; Smart Cities

1. INTRODUCTION

The use of data centers in public or private organizations has increased in recent years, mainly after the popularization of Cloud Computing. Monitoring of these environments has become an increasing necessity due to user demand and to a lack of energy supply [1] alongside associated problems of incorrect shutdown procedures [2]. Another problem that affects a data center is the failure of the cooling necessities required by the system, leading to issues controlling temperature and humidity. Anomalies within those two factors can increase the probability of hard data loss, error in writing, and reading data on hard disks, alongside RAM issues [3]. For this reason, the redundancy and monitoring of these services are fundamental [4].

An IoT approach with data analysis can supply this need, and our work explores an environment and power consumption sensor-based approach to detect temperature gradients compared with the associated global data center energy monitor process.

Our research group has been developing LoRa IoT solutions to monitor the environment in the building, and now we apply this approach to a data center where a working server affects the temperature in a rack environment. Gradient temperature was found with, sometimes, a 10ºC difference, there was also found correlations between temperature, energy consumption, and CPU usage.

2. RELATED WORK

In [5], the authors present a system called C-Oracle, which performs the thermal management of a data center environment. The work proposes policies for the identification of thermal emergencies. In its main policy, the authors make a check on the states of use of the data center in processing and storage.

Another work that approaches a similar theme and the one developed by J. Chen et al. [2]. In this study, the authors propose an intelligent sensor-and temperature control of data centers with a focus on energy efficiency. The system makes internal and external monitoring of the data center environment. This double monitoring allows the system to enable a ventilation system that controls the natural, expected the temperature of the data center based on relevant environment information. With this, it is possible to switch off some air-cooling system. However, this type of approach is not always practical and feasible, since some regions of the planet may inherit high temperatures throughout the year or over long periods. In this case, the authors present the design and evaluation of the system PTEC, a thermal and energy control prediction system in a data center. The focus of the study was to verify if the PTEC could efficiently perform monitoring of the data center with a low cost in the use of power. The system collects data from both the server fans and the air conditioner and checks whether the information on both temperature and consumption are within a margin of safety for the data center. The evaluation of the system is carried out using simulation to verify the efficiency and lability of the PTEC. The PTEC evaluation has shown that it can reduce cooling and circulation energy consumption by up to 34% and 30%.

In work done by N. Hassan at al. a Computational fluid dynamics analysis of airflow, temperature and pressure distribution of a university data center was done, where minimum air input temperatures of 18ºC and maximum air output of 31ºC were found to be normal and within standards but could be improved [6].

Finally, a similar work to this paper has been developed using an ESP8266 board and DHT11 temperature sensors, which then sends the data to a Ubidots cloud platform prepared to receive the data. This work consists of a Wi-Fi temperature and humidity monitoring system also applied to Data Centers. Similarly to other monitoring systems, it also has an alarm/alert functionality that will warn the owners in case any values has reached critical levels. [7]

Energy consumptions have also been one important ordeal that researchers have been trying to solve, and such, to try and reduce these consumptions researchers have tried to find a way to turn Data Center's Energy into Green Energy whose main objective is exactly that. In 2015 researchers created a project where they designed a

CCIOT 2019, September 20–22, 2019, Tokyo, Japan
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-7241-1/19/09…$15.00
DOI: https://doi.org/10.1145/3361821.3361824

green data center air conditioning system sourced on the auxiliary cloud system that was also used for data storage and big data analysis and prediction. This system not only controlled the air conditioning by switching it on and off based on the environment temperatures, but it would also send SMS alerts in case the temperatures reached dangerous levels. Overall they managed to reduce the Air Conditioning's energy consumption by 10% without reducing the data center's refrigeration effects [8].

It is also important to note that when supportive infrastructure energy consumption gets reduced, by shutting down the unnecessary, it is allowed a margin to increase the load by IT servers increasing performance. Research was done in legacy data centers where, with the use of electricity, temperature and humidity sensors, some researchers managed to achieve a 50% increase in IT load while shutting down unnecessary cooling systems[9], by managing to introduce a better air-flow, insulation and hardware water-cooling to improve results [10].

These last solutions aim to increase the Power Usage Effectiveness, or PUE for short, which describes how much energy is used in server hardware contrasting with cooling and other supportive systems.[11] In terms of the data center, energy consumption IT Servers and computing hardware will normally get two major consumers of energy, either the CPU or the RAM (Random Access Memory) depending on what the server was programmed to do. Google servers in 2007 would spend around 63% of the total consumption in CPU and DRAM processing only, while 10% was used in Disks and 5% on the network. Researchers in [12] also estimated that each server, depending on the hardware, could draw anything from 500W to 1400W, so a data center with over 20 servers could consume about 4900W to 5600W depending on the time of the day, in this case, servers were under-utilized but still consumed a sizable amount of power. This value may vary depending on the type of data center and what kind of hardware it may use [13].

The tech company HP presented on [14] an experiment with their new Dynamic Smart Cooling systems where 7500 sensors were deployed in each server rack to minimize energy consumption and maximize efficiency by redirecting cooling capacity to warmer areas inside the data center based on real-time readings. This approach requires a large investment for fully deployed/ working data centers, and large scale maintenance times to replace the outdated cooling system. It is also still one of the companies' projects in development and testing.

Finally, energy consumption monitoring alongside environment control can also be used as a less known use case: Disaster prevention and management. Researchers have found that accurate monitoring also allows increasing UPS efficiency that will help in extending the UPS battery time during a power outage and contributes to a more efficient facility design with a focus on cooling and air-flow which can, in time, increase the hardware's life [15].

The system presented by Sheikh Ferdoush et al. [16] attempts to solve problems caused by the use of components diffused in the IoT domain, such as rapid prototyping, in non-editable monitoring systems. The main-obstacles for these platforms to become a force for engineering, scientific and commercial domains consist of their lack of reliability, flexibility, scalability, interoperability, and difficulty of implementation, operation, and maintenance in the long term. This system solves by using a network of sensors developed with Arduino, Raspberry P, and XBee. The author states that this system has a low implementation cost and is highly scalable both in "relation to the types of sensors and about the number of sensors". The system makes internal and external monitoring of the data center environment. It is also stated that materials and methods should be described with sufficient details to allow others to replicate and build on published results. In our case the system was done solely using open sourced components and the code for the server and end-devices will be available on GitHub, the flow for the Dashboard in a Node-Red environment will also be provided in the GitHub for future improvements.

3. SYSTEM ARCHITECTURE

We can split the architecture into three main layers, Figure 1:

Sensor Layer: End Devices consist of the Environment Sensor integrated with a LoRa32u4 II board, and the Power Consumption Sensors integrated into the Arduino MKR WAN 1300 that enables data to be sent through a LoRa network into the closest gateway. End devices are also able to receive messages from the gateway. Therefore, communication is bidirectional

Network Layer: Consists of a LoRa Gateway connected to a Router that receives Uplinks from several end devices and send them to the TTN management platform. Then route those messages to one or more servers depending on the application they were defined to be sent to, while, also, forwarding messages from the servers to the respective End Devices, in case there is a need to send a command.

Knowledge Layer: The Flask web servers, a scalable solution, receives the data from the gateway and provides the treatment and processing of that data while also sending messages to certain sensors to calibrate their internal clock, it also provides the means to send alerts to the respective users when needed. The MariaDB databases provide the data storage and backup, all while Node-Red provides the dashboard that will display the data.

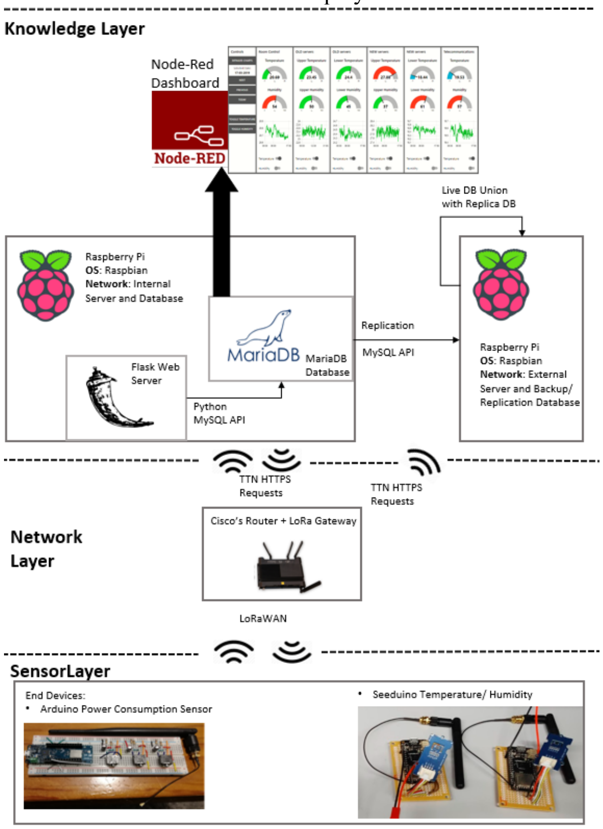


Figure 1. System Architecture developed

3.1 Sensor Layer

This layer is divided into: an environment sensor and the power consumption sensor.

Environment Sensor -The first step consists of prototyping the nodes responsible for collecting data from the environment in which they will be inserted.

Our prototypes see Figure 2, were created using LoRa32u4 II boards based on the Atmega32u4 microcontroller alongside a temperature and humidity sensor SHT31, whose accuracy was rated at values of ± 0.3 ° C. These prototypes are welded to circuit boards, and the entire circuit is placed inside a 3D printed box drawn in AutoCAD. As these sensors are designed to be portable and easy to place anywhere in the Data Center, they will be powered by 2400 mAh lithium batteries, whose autonomy is extended by code optimization and sleeps between submissions.

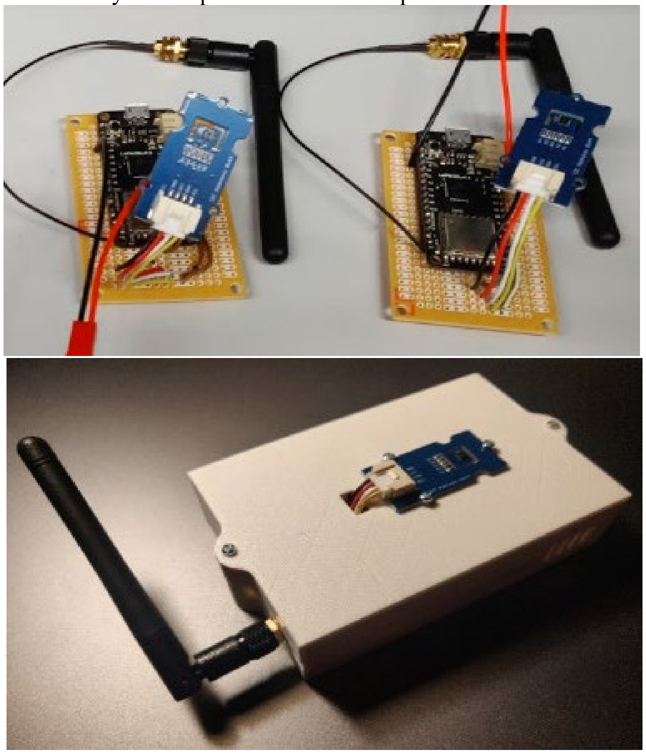


Figure 2. Developed sensor board (top) and Box drawn in AutoCAD and printed in lactic polyacid (PLA) in the FabLab a prototype laboratory of ISCTE-IUL

Power Consumption Sensor - The power consumption sensor, shown in Figure 3, was placed on the entry point that powers the server.

These prototypes use an Arduino MKR 1300 connected to current Transformers sensor with accuracies that will vary depending on the power consumption passing through the main cable at the time.

With values close to the expected rating of the current transformer we can expect a 3% error.

When measuring values that are very close to zero we can see errors up to 10%

Finally, if we go above the expected maximum rating by a large factor we risk not only burning the transformer, but we also get error results up to 70% due to saturation. [17]

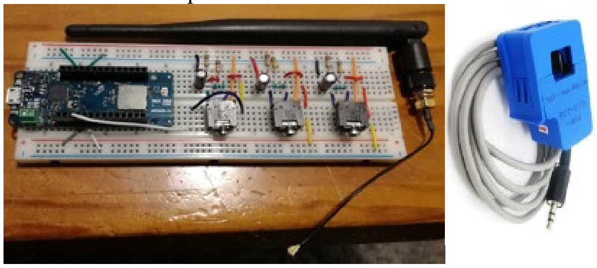


Figure 3. Developed Power consumption sensor board (left) and the Current Transformer (right) that connected to the jacks.

3.2 Network Layer

The entire system depends on two wireless communications protocols:

- The LoRaWAN protocol is used to establish a connection between the sensors and the LoRa gateway running on the Raspberry Pi.
- HTTPS is used to communicate between the LoRa gateway and the Application Server.

As a member of the LoRa Alliance, The Things Network, or TTN, is a worldwide project focused on developing open access and secure IoT network based on LoRa technology and LoRaWAN protocol. One of TTN's best feature is the possibility of creating a homemade Gateway with relatively cheap and easy to acquire hardware. The software setup is fully documented, and in a few hours, the Gateway is ready to receive messages sent from the sensors [18].

Each sensor is added to the network using the TTN console that provides a user-friendly interface to make device management as easy as possible. It also allows the integration of a variety of services such as HTTP. However, the Application server that will be used is going to connect directly to TTN.

3.3 Application Layer

Application Server

The Raspberry Pi is also where servers and applications run, these are, besides TTN gateway, Node-RED and Maria-DB. TTN gateway will be properly created using TTN guidelines.

The application server, created with python Flask Framework, is responsible for receiving the data from the LoRa Gateway server and processing it. The simplicity of the Flask framework [19], allows the creation of a server in a short time and with very little code, and the remaining lines correspond to the connection of the same to the MySQL database, installed in Raspberry processing of data collected in the database.

This server will have the function of processing the incoming HTTP messages with the UPLINK data packets from the devices and creating the corresponding objects according to the sensor type (since the method of encoding the data varies according to the manufacturer/programming), sending them to a database located on the same Raspberry PI.

The variables important for this paper would be Temperature, humidity, light, presence, air quality values, and finally, power consumption.

Finally, the server adds the functionality to send an Email to users, defined in the MySQL database

These emails are sent if any values become extremely anomalous, such as extreme temperatures detected, both higher and lower or current values that become too high which could be created by faulty hardware, or too low, since it can also mean that one or more servers have stopped working, or worse, the HVAC.

Dashboard for Data Center analysis

Node-RED is an open source software available for many OS, including the Raspbian [20]. It is a flow-based programming software, where it is possible to connect API's, physical devices, other network interfaces, and web services. It also allows the programming of rules in JavaScript to run or keep from running specific flows or the installation of extra features such as a TTN node that provides connectivity to TTN. Node-RED will receive its data from a local MySQL database where all the data is stored [21]; this way, communication between the Node-RED and the MySQL is all done internally. Therefore it's more secure and faster.

Node-Red will be used to visualize all the information created by the servers, it provides an easy graphical interface understandable by all types of users.

The dashboard allows the user to quickly detect anomalies on a specific server rack based on the temperature and humidity gauges that display real-time information, as the daily temperature charts that can be toggled to humidity. It also allows the user to visualize historical data by using the time controls on the left, see Figure 4

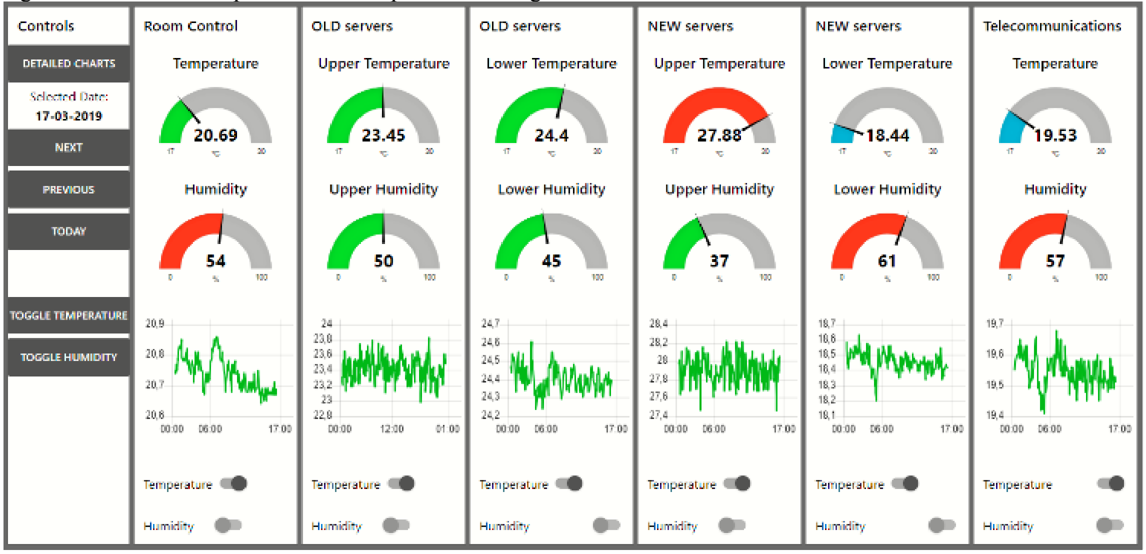


Figure 4. Dashboard Main Console, displaying details about each sensor

4. RESULTS

Data Center sensors were placed at specific points of interest, 2 of them on the newest server rack where most computing power is located, other two on the older server rack, one at the telecommunications rack and 1 Control sensor located on the outer room. The current transformer sensor was placed on the Room Control and read the summative power for all the servers, lights, and HVAC in the data center. Figure 5 shows its location.

After design, development, and implementation, the system ran for a month before any data pattern analysis was done, although, since it started gathering data, some issues were immediately detected.

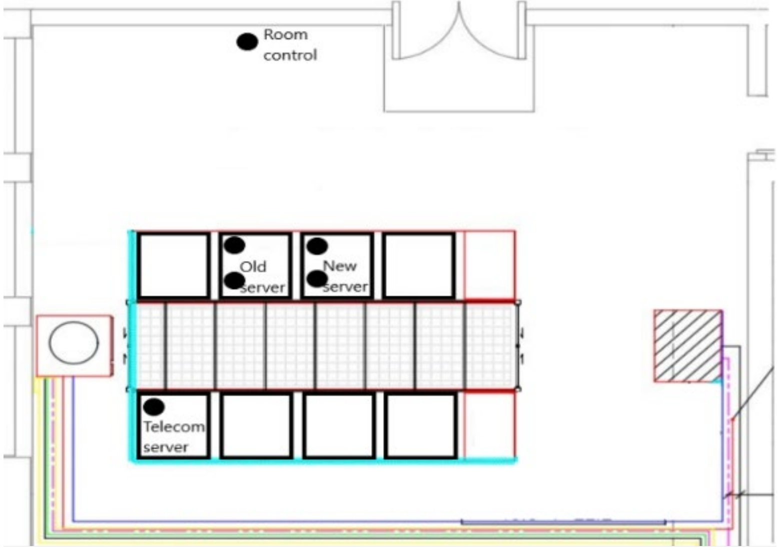


Figure 5. Data center sensors location

One of the first detected discrepancies would be the temperature difference inside the newest server rack where the sensor located at the upper shelf shows around 28°C while the lower shelf indicates 18°C, which provides us with a temperature variation of 10°C vertically. This variation was always constant for the project duration, and the current computing load can explain it these servers currently have, and the rack occupation itself that is near 80% which means less cold air flowing between the hardware.

The temperature gradient inside the older server rack is less than 1°C. However, the sensor on the upper shelf detects a slightly lower temperature than the lower shelf, as seen in Figure (6). Even if it seems contradictory, as the warmer air flows upwards, it can be explained by the rack occupation that is much lower than the newest rack and the upper shelves are almost empty which implies more air flow, sometimes colder than the temperature near the hardware on the lower shelf.

Figure (6) also shows that telecommunications servers have the lowest recorded temperatures with few or no fluctuations, and the room control sensor shows some fluctuations related to all server's current load. For this project's period and available data, no correlation between the rooftop above the data center and the room temperature was found.

The anomaly on Figure (6), in the second chart, was detected every day except Sundays, and the variation itself of less than 1°C wasn't considered a problem as the temperature of all the other sensors remains unchanged.

This unchanged temperature means the anomaly is not caused by the air conditioning system, but the hardware itself that emits more heat from 2 am to 3 am and heats up the outer room slightly.

The situation above was discussed with the IT department, and the concluded cause was a daily backup routine.

The anomaly mentioned in Figure (7) on all sensors was detected through the project period several times, but no pattern was found except for the duration of the anomaly and behavior. As it was detected on all sensors at the same time, it points to an air conditioning response to a temperature spike preceding the anomaly.

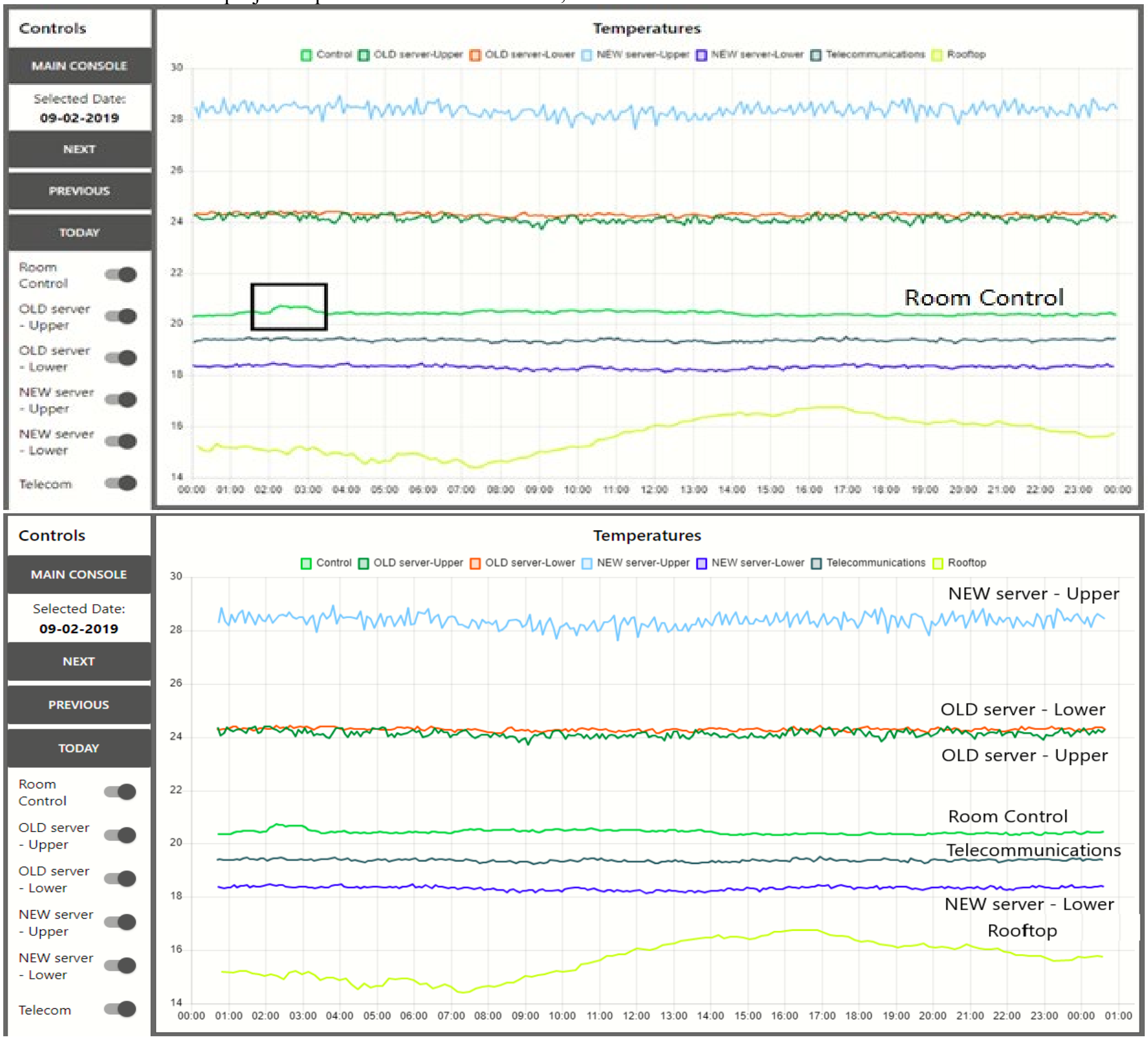
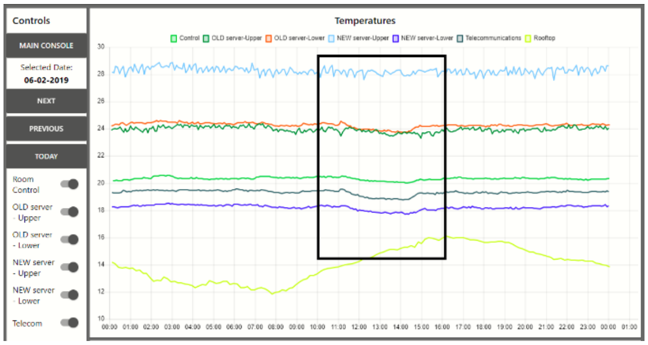
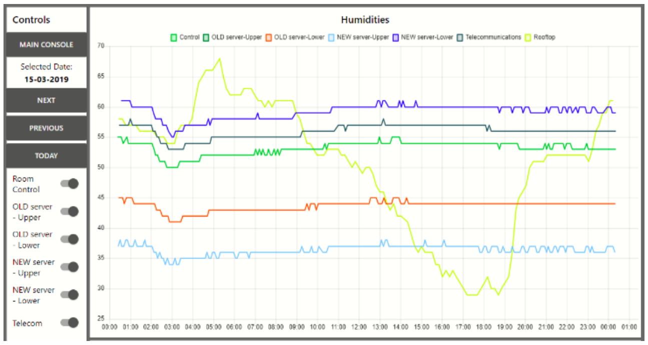


Figure 6. Dashboard Charts, displaying historical temperature readings for 09/02/2019 (top) and Data Center daily temperature variation pattern (bottom)



Selected Date: 06-02-2019

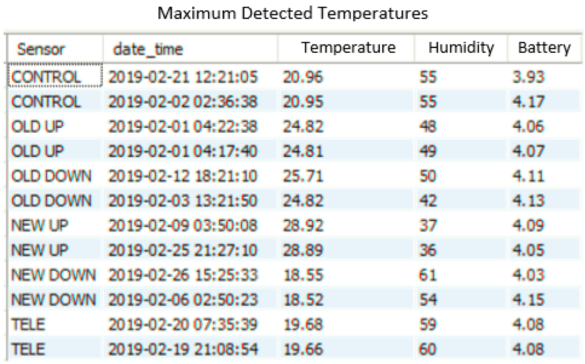
Figure 7. Data Center temperature variation anoma



Selected Date: 15-03-2019

Figure 8. Data Center humidity variation for 15/03/2019

Relating to humidity values, imprinted in Figure (8), there wasn't any visible correlation between the humidity outside the building, and inside the data center. Variations on these values were expected since humidity is related to temperature.



Maximum Detected Temperatures

Sensor	date_time	Temperature	Humidity	Battery
CONTROL	2019-02-21 12:21:05	20.96	55	3.93
CONTROL	2019-02-02 02:36:38	20.95	55	4.17
OLD UP	2019-02-01 04:22:38	24.82	48	4.06
OLD UP	2019-02-01 04:17:40	24.81	49	4.07
OLD DOWN	2019-02-12 18:21:10	25.71	50	4.11
OLD DOWN	2019-02-03 13:21:50	24.82	42	4.13
NEW UP	2019-02-09 03:50:08	28.92	37	4.09
NEW UP	2019-02-25 21:27:10	28.89	36	4.05
NEW DOWN	2019-02-26 15:25:33	18.55	61	4.03
NEW DOWN	2019-02-06 02:50:23	18.52	54	4.15
TELE	2019-02-20 07:35:39	19.68	59	4.08
TELE	2019-02-19 21:08:54	19.66	60	4.08

Figure 9. Data Center maximum and minimum detected temperatures

Maximum detected temperatures correspond to early morning backup routines and minimum detected temperatures to 1 pm – 2 pm period. From all sensors, both older and newer servers present bigger temperature variations of 2°C between maximum and minimum temperatures when room control and telecommunications server present only 1°C of maximum variation as proved by in Figure 9.

As for the power consumption values, presented in figure 10, the findings were in term with what was expected, during the day the values spiked up since HVAC routines create a requirement for power consumption and the overall processing power needed to deal with increased traffic also. At night, between 2 AM and 3 AM, the anomaly was verified when the power consumption increased slightly in the average value. HVAC is assumed to work with more intensity during work hours due to pre-defined routines and with less intensity during night time, and that is also one of the reasons there is an increase of temperature between 2 AM and 3 AM but with power consumptions that are still considered low when compared to day time consumptions.

Day time consumptions tend to be higher due to not only more consumption caused by the servers but also the HVAC has to work with more intensity during these times.

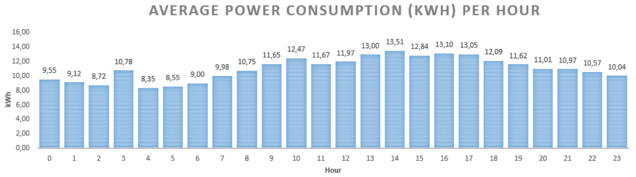


Figure 10. Data Center average power consumption by the hour.

5. DISCUSSION

When it comes to which system is better, obviously any system has its pros and cons. Unlike the PTEC [2] system and [6] which measures temperatures from the outside of the racks this system presents us with a compact solution that allowed us to measure the temperature at several rack levels and in real time.

Unlike the monitoring systems mentioned above, ours, while using LoRa can only send so many messages within one timeframe, but that timeframe allows for a close to real-time monitoring, which, complemented with the alarms from the server allows for faster disaster management in case an extreme anomaly is detected. LoRa is also known for its low power consumption, which allows for more battery autonomy when compared to connections such as Wi-Fi.

Our system is also created without any need for proprietary cloud platforms since its components are all open source, and the data is stored locally on raspberry PIs.

The ESP8266 prototype presented in [7] also infers a problematic issue, the DHT11, whose accuracy is valued at ± 2ºC, which can create anomalies where none exist. Our prototype, due to its use of SH31, has no such issues due to its more precise values, rated at ± 0.3ºC. Their prototype in [7] also requires the use of Wi-Fi which demands a gateway near and most likely a power source for it to last more than a few months, which in our system is not an issue.

Similarly mentioned in [6] we believe the data center could be improved with CoolDoor by CoolDoor Pty Ltd, a refrigerated rack door where the air being inputted into the racks would get chilled before entry. Thus it could improve energy consumption caused by the HVAC and, maybe, present better temperature values.

6. CONCLUSIONS

In this research work we apply a developed IoT solution and associated platform to a data center monitoring system, which is extremely important on these facilities where running costs, hardware maintenance and performance are a top priority since, as with many data centers, contains highly important information for the involved party. With the developed system, it is possible to determine anomalies and prevent future problems. This experiment proved that these anomalies are much more frequent than we initially thought and they can be caused by the hardware itself as much as the outdoor environment conditions. It was also possible to determine ways to improve the party's datacenter thermal flow by correctly managing the position of the server based on their usage, since servers with expected higher usage should have more free space around them, providing better airflow and reducing the temperature gradient that represents one of the most important factors on hardware durability.

7. ACKNOWLEDGMENTS

This work has been partially supported by Portuguese National funds through FITEC programa Interface, with reference CIT "INOV - INESC INOVAÇÃO - Financiamento Base".

8. REFERENCES

[1] T. Lu, X. Zha, and X. *Zhao*, “Multi-stage Monitoring of Abnormal Situation Based on Complex Event Processing,” *Procedia Comput. Sci.*, vol. 96, pp. 1361–1370, 2016.

[2] J. Chen, R. Tan, G. Xing, and X. Wang, “PTEC: A System for Predictive Thermal and Energy Control in Data Centers,” in *2014 IEEE Real-Time Systems Symposium*, 2014, pp. 218–227.

[3] N. El-Sayed, I. A. Stefanovici, G. Amvrosiadis, A. A. Hwang, and B. Schroeder, “Temperature management in data centers: why some (might) like *it* hot,” *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 1, pp. 163–174, 2012.

[4] Q. Liu, Y. Ma, M. Alhussein, Y. Zhang, and L. Peng, “Green data center with IoT sensing and cloud-assisted smart temperature control *system*,” *Comput. Networks*, vol. 101, pp. 104–112, 2016.

[5] L. Ramos and R. Bianchini, “C-Oracle: Predictive thermal management for data centers,” in *2008 IEEE 14th International Symposium on High Performance Computer Architecture*, 2008, pp. 111–122.

[6] N. M. S. Hassan, M. M. K. Khan, and M. G. Rasul, “Temperature Monitoring and CFD Analysis of Data Centre,” *Procedia Eng.*, vol. 56, pp. 551–559, 2013.

[7] S. Saha and A. Majumdar, “Data centre temperature monitoring with ESP8266 based Wireless Sensor Network and cloud based dashboard with real time alert system,” in *2017 Devices for Integrated Circuit (DevIC)*, 2017, pp. 307–310.

[8] Z. Zhang, C. Lai, and H. Chao, “A green data transmission mechanism for wireless multimedia sensor networks using information fusion,” *IEEE Wirel. Commun.*, vol. 21, no. 4, pp. 14–19, 2014.

[9] D. Edgar and R. Pfeifer, “Power Management in a Legacy Data Center,” 2011.

[10] M. Bennett, “IoT – Sensors for Data Center Management.”

[11] B. Kleyman, “Why monitoring data center power consumption is vital,” 2012.

[12] M. Dayarathna, Y. Wen, and R. Fan, “Data Center Energy *Consumption* Modeling: A Survey,” *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 732–794, 2016.

[13] A. Vasan, A. Sivasubramaniam, V. Shimpi, T. Sivabalan, and R. Subbiah, “Worth their watts? - an empirical study of datacenter servers,” in *HPCA - 16 2010 The Sixteenth International Symposium on High-Performance Computer Architecture*, 2010, pp. 1–10.

[14] J.-P. Kamath, “HP shows smart cooling boosts datacentre savings,” *Comput. Wkly.*, 2007.

[15] J. S. Klaus, “Disaster Antidote: Getting Control of Data Center *Power*,” *Enterp. Syst. J.*, 2012.

[16] S. Ferdoush and X. Li, “Wireless Sensor Network System *Design* Using Raspberry Pi and Arduino for Environmental Monitoring Applications,” *Procedia Comput. Sci.*, vol. 34, pp. 103–110, 2014.

[17] R. Wall, “YHDC SCT-013-000 Current Transformer Report.” [*Online*]. Available: https://learn.openenergymonitor.org/electricity-monitoring/ct-sensors/yhdc-sct-013-000-ct-sensor-report. [Accessed: 01-Feb-2019].

[18] “The Things Network. (2019). Build your own gateway.” [Online]. *Available*: https://www.thethingsnetwork.org/docs/gateways/start/build.html. [Accessed: 03-May-2019].

[19] Flask.pocoo.org, “Quickstart — Flask 1.0.2 documentation.,” 2019. [Online]. Available: http://flask.pocoo.org/docs/1.0/quickstart/. [Accessed: 13-Feb-*2019*].

[20] Nodered.org, “Node-RED : Running on Raspberry Pi,” 2019. [Online]. Available: https://nodered.org/docs/hardware/raspberrypi. [Accessed: 25-Feb-2019].

[21] Flows.nodered.org, “node-red-node-mysql,” 2019. [Online]. *Available*: https://flows.nodered.org/node/node-red-node-mysql. [Accessed: 02-Mar-2019].