RZ 3807 (# Z1107-001) 07/04/2011
Electrical Engineering 14 pages

# Research Report

## Wireless Sensor Network for Continuously Monitoring Temperatures in Data Centers

Beat Weiss, Hong Linh Truong, Wolfgang Schott, Thomas Scherer, Clemens Lombriser, and Pierre Chevillat

IBM Research – Zurich
8803 Rüschlikon
Switzerland

# Wireless Sensor Network for Continuously Monitoring Temperatures in Data Centers

Beat Weiss, Hong Linh Truong, Wolfgang Schott,
Thomas Scherer, Clemens Lombriser, Pierre Chevillat

IBM Zurich Research Laboratory (ZRL)
Rüschlikon, Switzerland

***Abstract*—Continuous monitoring of the spatial temperature distribution in a data center (DC) is important for reliably operating the computing equipment and minimizing the required cooling energy. For this purpose, we track the temperatures in thermally critical areas of the DC with low-cost sensors and forward the captured information via the ZRL Data Center Wireless Sensor Network (DCWSN) to an application for generating interpolated temperature maps. The proposed solution has been successfully installed and deployed at the IBM Data Center in Geneva, Switzerland.**

## I. Introduction

The successful operation of a large-scale air-cooled data center (DC) requires an efficient cooling environment to ensure that the DC operator can provide its services to customers with maximum availability and reliability at minimal operational cost. An efficient cooling system guarantees that the temperatures at the inlets of all computing devices in the DC never exceed a given threshold value to prevent device overheating, and achieves this goal with minimum required cooling energy. However, most of today's DCs waste cooling energy because they are operated at significantly lower temperatures than actually necessary. This approach certainly reduces the potential risk of reacting too late to a harmful temperature increase in the DC that might be caused by a sudden malfunction or replacement of a cooling or computing device, but leads to the consumption of 3 to 5 % more cooling energy for each degree Celsius below the upper temperature limit. To encourage operators to run their DC closer to the economically attractive upper limit, monitoring functions have to be incorporated into existing DC management tools that allow for continuous tracing and analyzing of the temperature distribution in the DC during its regular operation.

The temperature distribution in the DC can be obtained by using IBM's Mobile Measurement Technology (MMT) [1]. This technology allows for rapid and systematic gathering of 3-dimensional temperature information with a mobile cart and visualizing the captured data in temperature maps. MMT has been often applied in air-cooled DCs with racks forming hot and cold aisles to identify hot spots and cooling inefficiencies. Based on the survey results, the undesired thermal effects could be remedied by changing the airflow from the water-cooled computer room air conditioners (CRACs) through the raised-floor plenum and perforated floor tiles to the inlets of the servers in the racks. Since the cart typically carries up to 100 spatially distributed temperature sensors and can be readily moved from tile to tile, MMT can provide high-resolution temperature maps of the DC; however, these maps only represent snapshots of the temporally changing temperature distribution in the DC.

Continuous monitoring of the temperature distribution in the DC can be performed by placing low-cost temperature sensors at some key locations and forwarding the information from the sensors via relay nodes and a gateway to a monitoring client that generates and iteratively updates interpolated temperature maps of the DC. Several commercially available wireless sensor networks can be used to implement such a real-time temperature monitoring solution (e.g. SynapSense [2], Federspiel [3], DustNetworks [4]). We propose to use the ZRL Data Center Wireless Sensor Network (DCWSN) solution that is tailored to the needs of the DC monitoring application. In the DCWSN, all nodes are battery-powered, equipped with an IEEE 802.15.4 compatible radio transceiver [5], and communicate with each other by executing the IMPERIA protocol stack on an IRIS-Moterunner processing platform [6]. The stack performs all required network functions, uses the publish/subscribe messaging protocol MQTT-S [7] for communicating between sensor nodes and the monitoring application, and can be easily integrated with IBM's Tivoli Monitoring software.

The ZRL DCWSN solution has been successfully installed and deployed at the IBM Data Center in Geneva, Switzerland. It is a production DC with a raised floor area of about 2,200 $m^2$ that houses 400 racks with heterogeneous computing equipment and is cooled by 40 CRACs and 2 chillers. The temperature changes in the cold aisles of the DC were tracked with 108 sensors from March 23 to April 26, 2011. During this period, the operator stepwise upgraded the DC cooling system to reduce its energy consumption from 3,600 MWh to 1,500 MWh per year. This was achieved by replacing the chillers, exploiting free cooling, and increasing the cold water and room temperature from 8 °C to 13 °C and 21 °C to 24 °C, respectively. Since the regular DC operation could not be interrupted during the upgrade, continuous monitoring of the temperature distribution in the cold aisles of the DC was very important to timely detect potentially harmful temperature increases at the server inlets, and immediately combat them

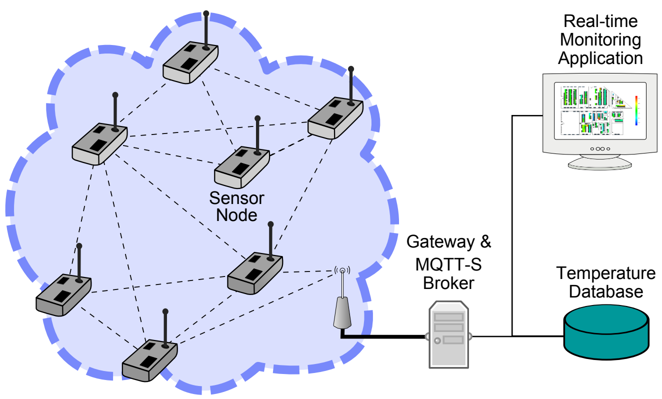



Fig. 1. The DCWSN solution comprises the sensor nodes that communicate with each other by executing the IMPERIA protocol stack, a gateway managing the network and hosting the MQTT-S broker, and a client application for real-time monitoring and data logging.


Fig. 2. Wireless sensor node mounted on an IBM server rack to measure the inlet temperature. Magnetic mounting allows for fast and easy deployment.

by moving and replacing perforated tiles in the DC to improve the airflow and temperature distribution.

This report describes the measurement results obtained with the ZRL DC Wireless Sensor Network at the IBM Data Center in Geneva. In Section 2, an overview on the DCWSN solution is given. Section 3 presents the results by discussing network performance issues and interpreting various obtained temperature graphs. Further results can be found in the Appendix. The report concludes with a brief summary.

## II. The DC Wireless Sensor Network Solution

Figure 1 shows the architecture of the ZRL DCWSN solution. The DCWSN is composed of a number of battery-driven wireless temperature sensor nodes placed at key locations in the DC, e.g. at the front of selected server racks as shown in Figure 2. Each of those sensor nodes periodically measures the local temperature and publishes it by using the MQTT-S interface of the IMPERIA protocol stack. IMPERIA forwards the data directly or via other sensor nodes to the MQTT-S broker that runs on the gateway. To receive measurement data, a client application can subscribe to these data at the MQTT-S broker, which then forwards the temperature values received from the DCWSN to the subscriber.

The IMPERIA protocol stack implements a centrally managed low-power multi-hop wireless network operated on a time division multiple access (TDMA) scheme [6]. The controller on the gateway manages the network in two modes: a management mode and a synchronized mode. In the management mode, all sensor node radio transceivers are constantly in receive mode. The gateway issues commands to instruct individual nodes to report their local network topology and to measure link qualities. With the collected global network topology information, the gateway computes a routing tree and a TDMA schedule for the sensor nodes. The schedule defines time slots, in which each sensor node can either send data to its parent node, receive data from its children, or turn off its radio transceiver to save power. As the schedule only allocates a single sender to each slot, no interference from other senders has to be expected, and the data can efficiently be transported through the network.

After configuring the schedule on the sensor nodes, the gateway establishes the synchronized mode. Now, the DCWSN runs in low-power mode, and the sensor nodes only enable their radio transceivers within their assigned slots to either send or receive data. The schedule is periodically repeated to deliver updated temperature data to the MQTT-S broker on the gateway.

We have chosen the MQTT-S messaging protocol as middleware solution as it has been tailored to the low-power needs of sensor networks and can also run over a wide range of network solutions, such as cellular and local area networks. MQTT-S follows a data-centric communication paradigm where publishers assign topics to their data when they publish it to the broker. Subscribers, on the other hand, specify the topics on which they want to receive data. This data-centric communication approach decouples publishers from the subscribers and simplifies the requirements on the protocol stack on the sensor nodes since all communication flows via the broker.

The ZRL DCWSN solution includes a real-time monitoring application that subscribes to temperatures measured by the DCWSN and visualizes the collected data by generating interpolated temperature maps of the DC. All measurement data collected by the monitoring application are stored in a database for reference and later analysis.

## III. Measurement Results

We have deployed our DCWSN solution at the IBM Data Center in Geneva. This DC consists of two modules with a raised floor area of about 2,200 $m^2$ and a total of 20 rooms, as shown in Figure 3. A third module is currently under construction. Most of the 400 racks in the DC are arranged such that they form hot and cold aisles, with some of the areas featuring a cold aisle containment. To further improve the cooling efficiency in the DC, most of the racks will be

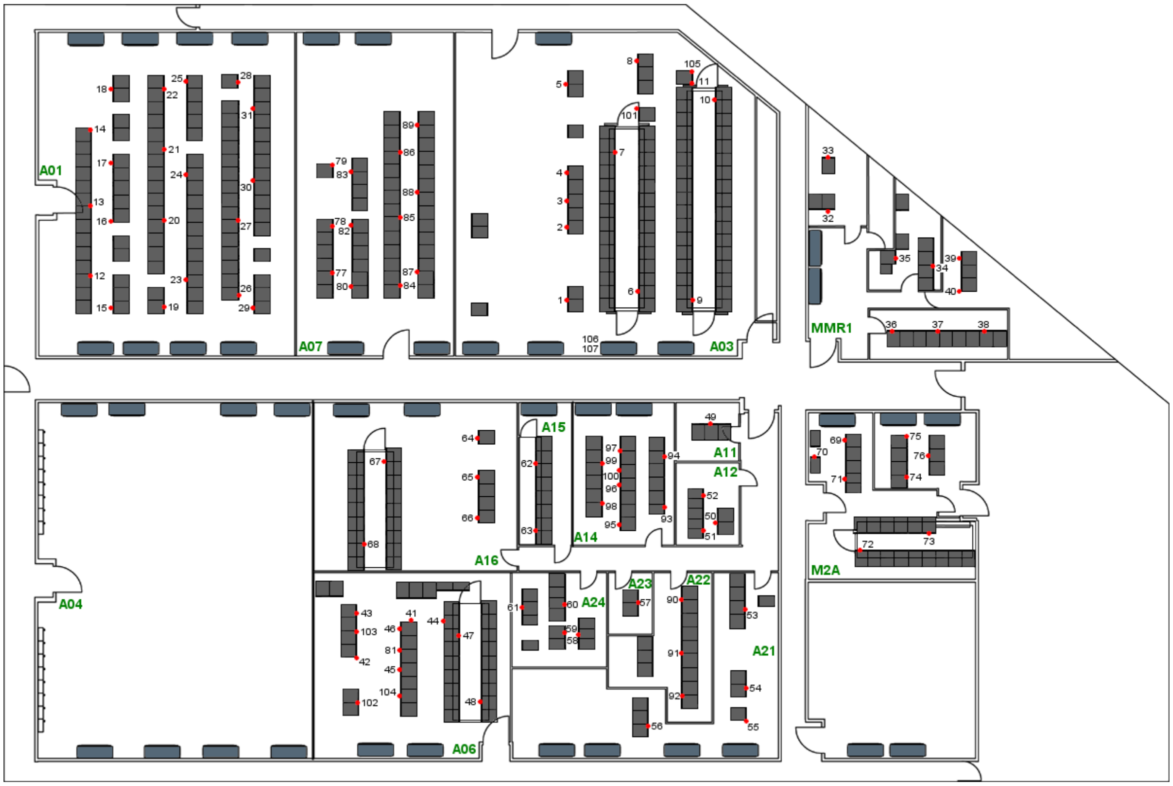



Fig. 3. Map of the IBM Data Center in Geneva. 108 sensor nodes were installed to measure the temperature at the rack inlets in the cold aisles.

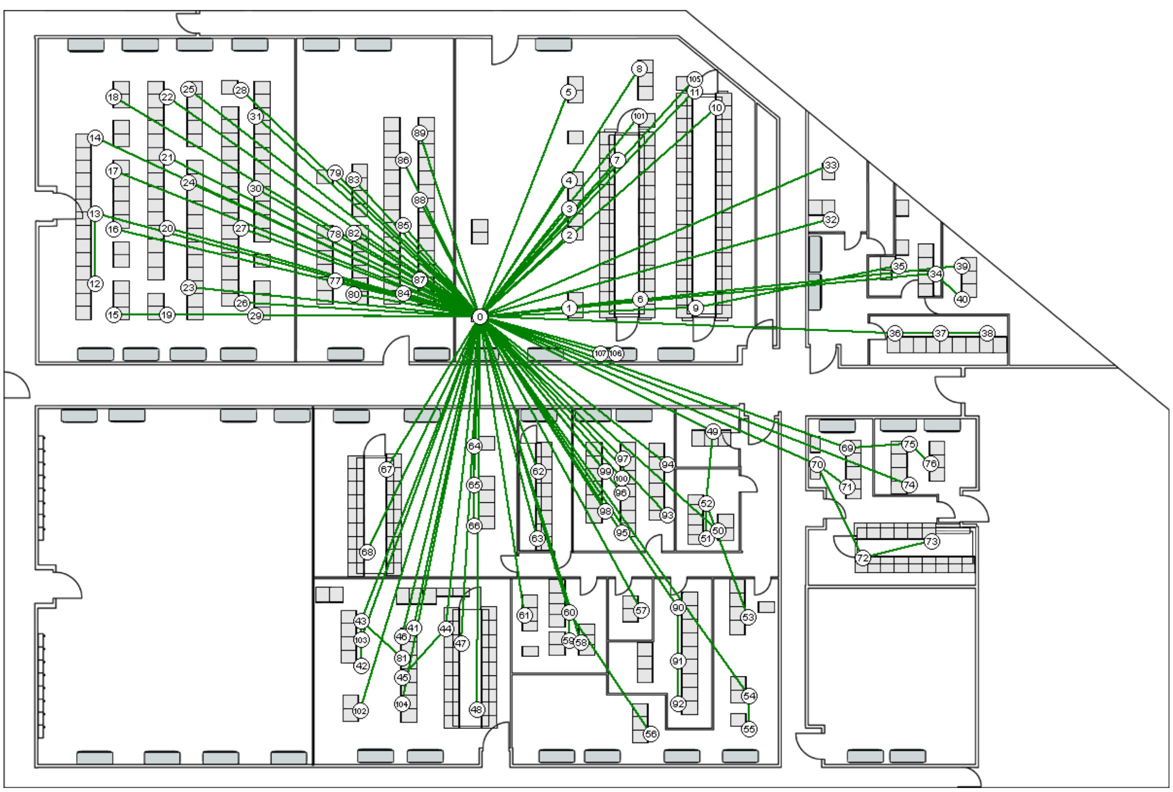


Fig. 4. Routing tree used for the DCWSN in the IBM Data Center in Geneva. The network has mostly a star topology, except for a few nodes that were hard to reach directly.

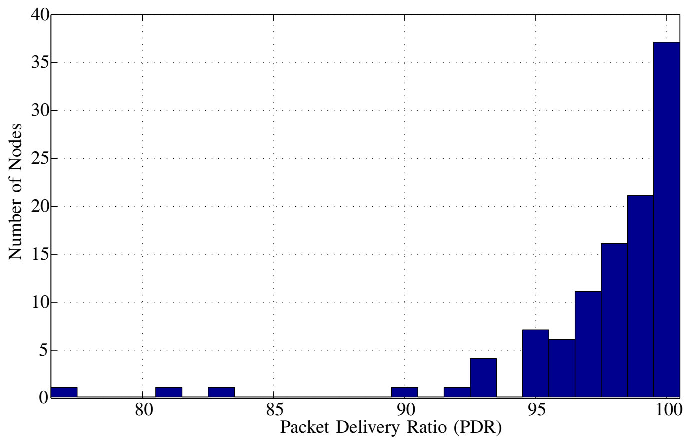



Fig. 5. Histogram showing the number of nodes reaching a certain PDR. The mean PDR of all nodes is 97.63 %.

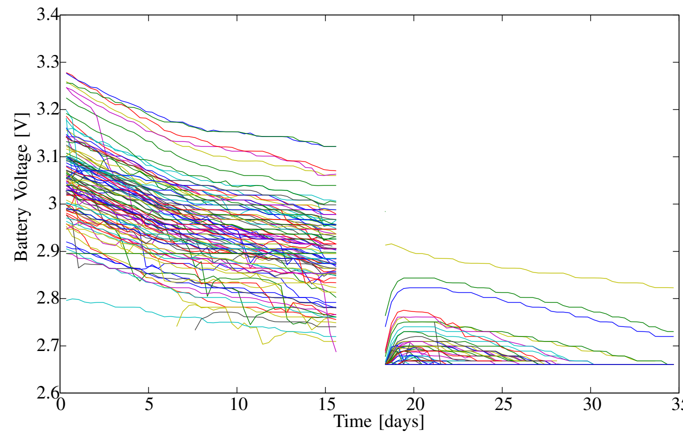



Fig. 6. Sensor node battery voltage trace over the 5 weeks runtime. During the 3-day connectivity loss, the sensor nodes operated in the management mode, using considerably more energy and shortening the network lifetime.

covered with a cold aisle containment in the near future.

We installed the DCWSN with 108 temperature sensor nodes to continuously monitor the temperature in the DC during the chiller upgrade and raising of the chilled water temperature. The sensor nodes were attached to the front side of the racks at a height of 1.6 m to measure the inlet temperatures. We used these data to generate interpolated temperature maps of the cold aisles in the entire data center. Each sensor node was assigned an identification number between 0 and 107. The sensor locations are shown in the floor plan in Figure 3. In this section, the sensor nodes will be referenced by their respective identification number and room number (A1,. . . , A24, MMR1 and M2A), as indicated on the floor plan.

### A. *Network performance*

After having deployed all DCWSN nodes, the gateway started its network topology discovery process to detect and assess the quality of the wireless links between the sensor nodes. We discovered that all nodes were directly reachable by the gateway, which was equipped with an antenna providing a 3 dB stronger gain than the ones used by the sensor nodes. The network was well connected with many good links, indicating a wireless-friendly environment. Due to the good results, we initially chose a star topology, where each sensor node was assigned one TDMA slot to send its messages directly to the gateway. The IMPERIA schedule applied in the synchronized mode thus included one slot to broadcast synchronization beacons from the gateway to all sensor nodes and 107 slots to collect the data from each node. Using a slot duration of 30 ms, the total data collection time amounted to about 3 seconds. The schedule and hence the data collection was repeated every 10 s. In the synchronized mode, each node thus had to periodically enable its radio transceiver for one slot to receive the synchronization beacon and one further slot to send its data, resulting in a duty cycle of 0.6 %.

The network ran for 35 days and collected 29 million temperature samples. A failure at the gateway led to a network outage of 3 days, in which the sensor operated in the management mode, thus running at 100 % duty cycle instead of just 0.6 %. The network outage unfortunately started at the beginning of a weekend and could thus only be discovered at the next working day. The failure was due to a crash of the USB driver handling the USB connection between the gateway computer and the sensor node that provides the gateway with a connection to the DCWSN. This connection could be restored through a remote reset of the USB driver, after which the synchronized mode could quickly be re-established because the network topology information was still available at the gateway.

To assess the reliability of the DCWSN with a star network topology, we collected statistical data such as the packet delivery ratio (PDR). The PDR is given by the number of messages successfully received by the gateway divided by the number of messages transmitted by a sensor node. Figure 5 shows a histogram of the PDR achieved by the sensor nodes. The mean PDR was 97.63 %, and only 3 sensor nodes had a PDR lower than 90 %. Note that the computed PDR excludes the 3-day network outage as no messages were received by the gateway at that time.

During the first week, we monitored the PDR and made adjustments to the topology where needed. Sensor nodes with a low PDR were reconfigured to send their data to the gateway via a nearby relay node that had a better PDR than the source node. The final network topology for the remaining measurement is shown in Figure 4.

The power consumption of the network was traced by measuring the battery voltage of the sensor nodes. Figure 6 shows the voltage traces that initially dropped exponentially but then flattened out until the network outage. This is a typical behavior for alcaline batteries, and we would expect the voltage to remain stable in the subsequent days and weeks. During the outage, however, the voltage dropped significantly faster because the nodes were operated with a 100 % duty cycle. After re-establishing the synchronized mode, the battery voltage increased again since IMPERIA's duty cycle is low enough to allow battery relaxation during the sufficiently long

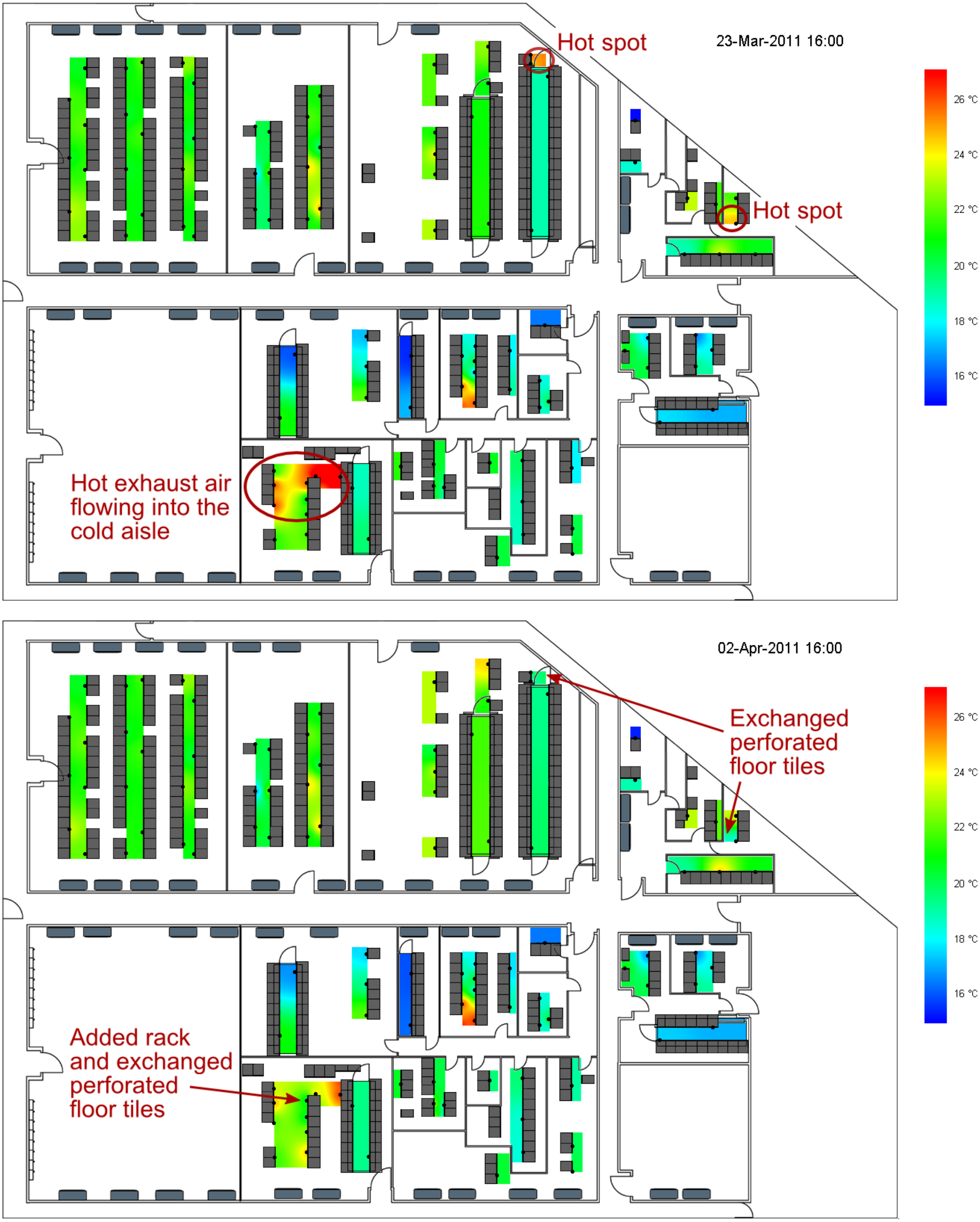



Fig. 7. Interpolated temperature maps of the cold aisles on the first day of the measurement campaign (top) and 10 days later after exchanging perforated floor tiles and improving the airflow distribution (bottom).

Fig. 8. Perforated floor tile was exchanged by one with bigger open area to provide more cold air to the rack at the location of node 40.

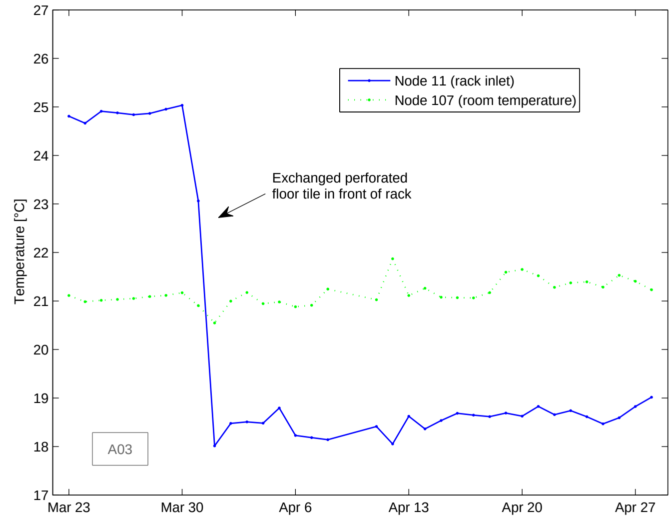



Fig. 9. Perforated floor tile in front of rack with temperature sensor node 11 was exchanged on March 31 by one with bigger open area. This yielded a reduction of more than 6 °C of the air inlet temperature at the top of the rack. The plot depicts the daily average temperature measured at the rack inlet and at the intake of a computer room air conditioner in the same room.

time period between use [8].

We could not assess the lifetime of the DCWSN during its deployment at the IBM Data Center in Geneva because the batteries of the sensor nodes were not fully depleted during the measurement period, and due to the network outage caused by the USB driver as can been seen in the battery voltage traces shown in Figure 6. Moreover, three sensor nodes stopped operating during the DCWSN deployment, caused by a software bug. Even though the nodes could re-join the network and continued reporting measurements to the gateway after a simple manual reset, the increased energy consumption while temporarily operating in the management mode could not be taken into account in the network lifetime assessment. In addition, inaccurate lifetimes were obtained because the operating system abstracts the measured battery voltage into an internal battery value indicator with a minimum value of 2.66 V. The battery voltage traces show that a number of sensor nodes were running at this minimum value for more than a week, indicating that IMPERIA can still operate the sensor nodes at much lower battery voltage. In short, Figure 6 only indicates that the network can run at least 35 days.

To assess the lifetime of the DCWSN, we are currently measuring the battery lifetime of the sensor nodes in a separate testbed with a comparable network topology and schedule. In this testbed, the sensor nodes have been continuously running in the synchronized mode for 8 months, and based on the measured voltage trace, we expect to reach a total lifetime of 32 months using a 3 Ah battery. Even longer battery lifetimes could be reached by increasing the interval between two measurements from 10 seconds to several minutes.

### *B. Temperature measurements*

We visualized the measurement data with interpolated temperature maps of the DC areas that were covered by the DCWSN. Such a temperature map of the IBM Data Center in Geneva is shown in Figure 7. For this setup, we have defined one interpolation zone per cold aisle. The temperature values of all sensors in an interpolation zone are weighted with the inverse quadratic Euclidean distance from the sensor location to the respective grid point. Areas where no temperature information is available remain white. The interpolation algorithm works well for small interpolation zones with only a few sensor nodes per zone and is thus a good choice for cold aisle monitoring with a limited number of sensors.

Figure 7 shows the interpolated temperature map of the cold aisles in the IBM Data Center in Geneva on the first day of the measurement campaign and 10 days later after some of the perforated floor tiles have been exchanged to improve the distribution of the cooling air. The temperature measurements in the cold aisles are mostly below 25 °C. One exception is room A06, where hot exhaust air is flowing directly into the cold aisle on the left. This yields high inlet temperatures near nodes 42, 43, 46 and 103. Other hot spots are identified near nodes 11 and 105 in room A03, and near node 40 in room MMR1. The hot spot in room A14 was ignored because there was no critical equipment in operation at this particular rack location.

Adding new perforated floor tiles or replacing existing ones by floor tiles with bigger open area is an easy way to provide more cold air to areas with critical temperature, given that the pressure level in the sub floor plenum is sufficient. By exchanging the perforated floor tile in front of node 40 by one with a bigger open area to provide more cold air to the rack, the inlet temperature could be reduced by more than 10 °C. Figure 8 shows the new perforated floor tile in front of the rack. The perforated floor tile in front of nodes 11 and 105 was also exchanged on March 31 by one with bigger open area. This yielded a reduction of 6.5 °C of the air inlet temperature

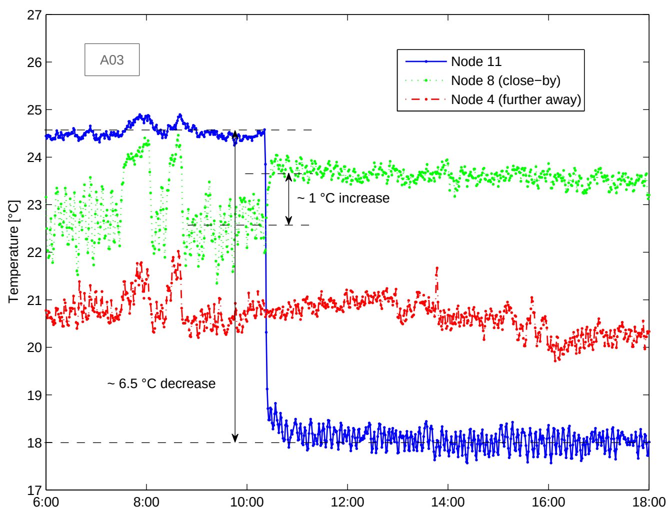



Fig. 10. Detailed view of temperature measurements on March 31. While the temperature at node 11 decreased by 6.5 °C, node 8 (which was attached to a rack in a neighboring aisle) measured a temperature increase of 1 °C. Node 4 (which was attached to a further away rack) did not measure a temperature increase.

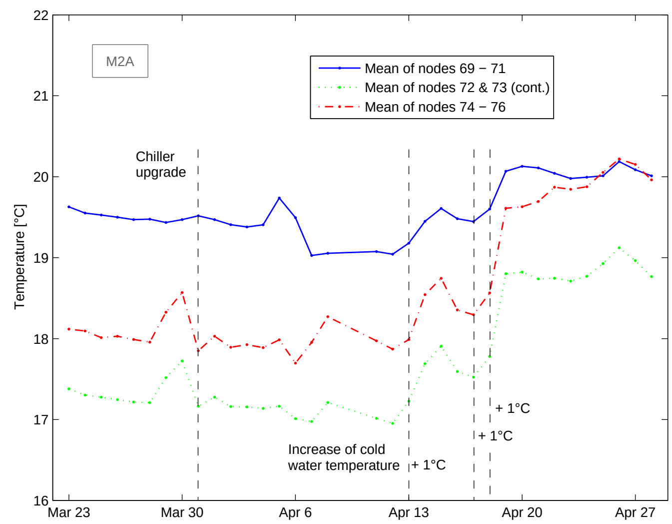



Fig. 11. Daily average temperature measurements for all sensor nodes in room M2A. There is a significant temperature increase after the cold water temperature was raised by 3 °C.

at the top of the rack. The overall room temperature was not affected by this action, as is depicted in Figure 9. A detailed view of the temperature measurements on March 31 is shown in Figure 10. While the temperature at node 11 decreased by 6.5 °C, node 8, which was attached to a rack in a neighboring aisle, measured a slight temperature increase. This can be explained by the pressure drop in that particular area of the sub floor plenum due to the higher airflow through the new perforated floor tile. Node 4, which was attached to a rack further away from the new perforated floor tile, did not sense a temperature increase. In room A06, a rack was moved to the location of node 41 to better shield the cold aisle from the hot exhaust air. Additionally, some perforated floor tiles with big open area were added. As shown in the lower map of Figure 7, the hot spots could successfully be eliminated by these actions.

Figure 11 shows the temperature in room M2A. The chillers were replaced on April 1, and the chilled water temperature was raised in 3 steps from 8 °C to 11 °C on April 13, 17, and 18. As expected, the chiller upgrade had no significant effect on the temperature in the DC, but rising the cold water temperature led to a substantial increase of the rack inlet temperature in this particular room. However, some other areas in the DC were not affected as much. Temperature measurements of all sensor nodes are included in the Appendix. During the measurement campaign, with a 3 °C increase of the chilled water temperature, the average temperature over all sensor nodes was increased by only 0.7 °C. This means that the CRACs were able to compensate part of the higher cold water temperature by increasing the airflow.

## IV. Conclusion

The spatial temperature distribution in an energy-efficient, air-cooled DC has to be continuously monitored to prevent both overheating of computing equipment as well as wasting of cooling energy. To this end, we installed the ZRL DCWSN in the DC to capture the temperature at some key locations with low-cost battery-powered sensors, forward the captured information from the sensors via relay nodes to a monitoring client using networking functions of the IMPERIA/MQTT-S protocol stack, and iteratively generate interpolated temperature maps on the client computer. We successfully deployed the proposed real-time monitoring solution at the IBM Data Center in Geneva to trace the temporally changing temperatures in the DC during a cooling system upgrade that has been performed by the DC operator to reduce the energy consumption of the DC. During the five weeks measurement period, the DCWSN performed reliably. Several potentially harmful hot spots were detected and could be eliminated by improving the airflow and temperature distribution in the room. Due to the successful deployment of the ZRL temperature monitoring solution, the DC operator considers to permanently install the DCWSN at its site to minimize the increased risk of overheating servers in the new energy-optimized operating environment.

## References


[1] H. Hamann, J. Lacey, M. O'Boyle, R. Schmidt, and M. Iengar, "Rapid Three-Dimensional Thermal Characterization of Large-Scale Computing Facilities," in *IEEE Transactions on Components and Packaging Technologies*, 2009.

[2] SynapSense, "Wireless Environmental Monitoring and Energy Management." [Online]. Available: http://www.synapsense.com

[3] Federspiel, "Energy Management Systems for Data Centers." [Online]. Available: http://federspielcorporation.com

[4] DustNetworks, "Intelligent Wireless Sensor Networks." [Online]. Available: http://www.dustnetworks.com

[5] ANSI/IEEE, "IEEE LAN MAN Standards, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," 2006.

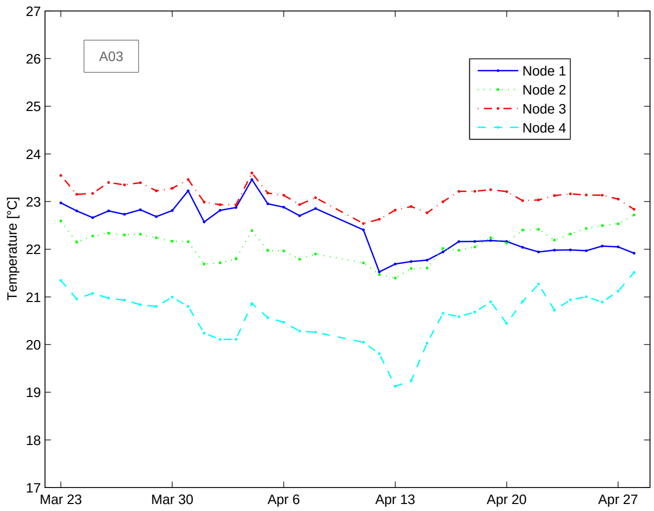
[6] B. Weiss, H. L. Truong, W. Schott, A. Munari, C. Lombriser, U. Hunkeler, and P. Chevillat, "A Power-Efficient Wireless Sensor Network for Continuously Monitoring Seismic Vibrations," in *8th IEEE Communications*

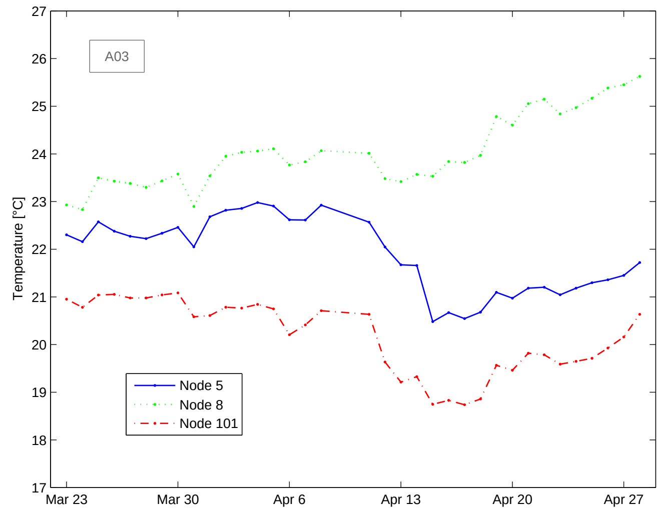
*Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON)*, 2011.

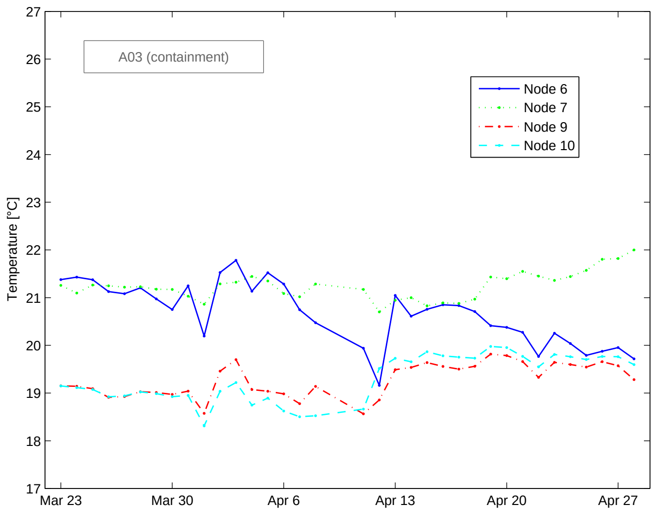
[7] U. Hunkeler, H. Truong, and A. Stanford-Clark, “MQTT-S: A Publish/Subscribe Protocol for Wireless Sensor Networks,” in *Proc. Workshop on Information Assurance for Middleware Communications (IAMCOM)*, 2008.

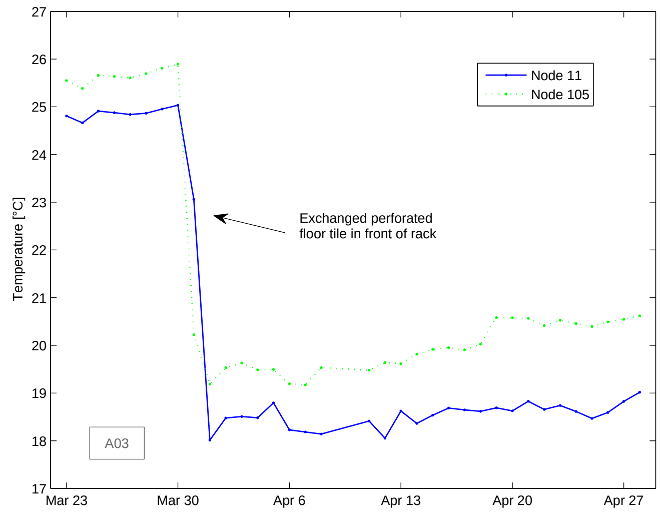
[8] C.-K. Chau, M. H. Wahab, F. Qin, Y. Wang, and Y. Yang, “Battery recovery aware sensor networks,” in *7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*. IEEE, Jun. 2009, pp. 1–9. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5291623

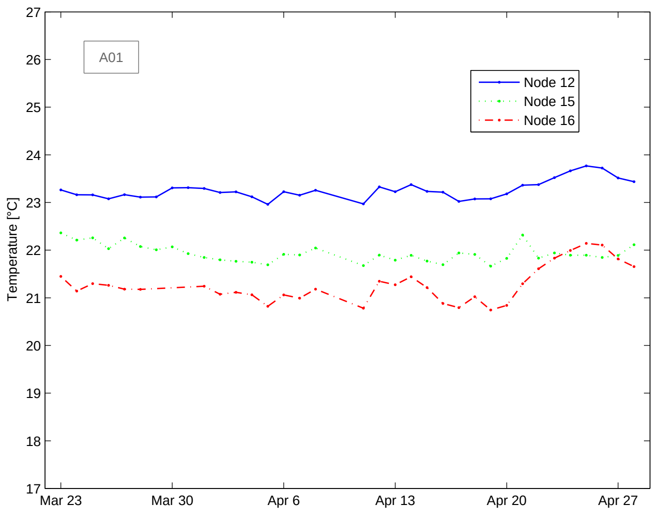
## Appendix
## Daily Average Temperature at Individual Sensor Nodes

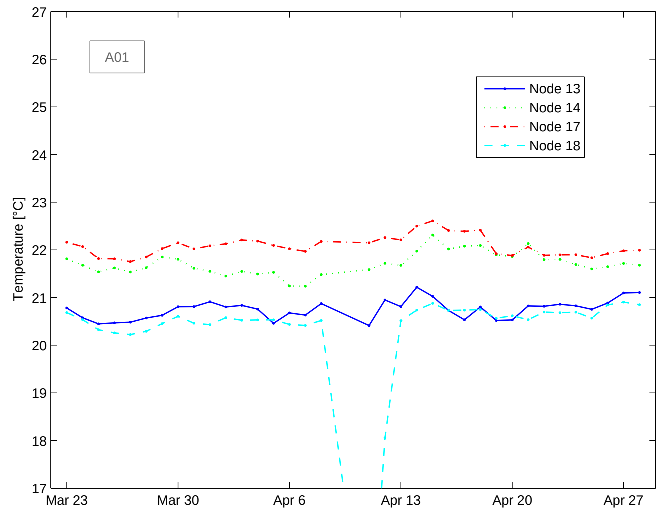


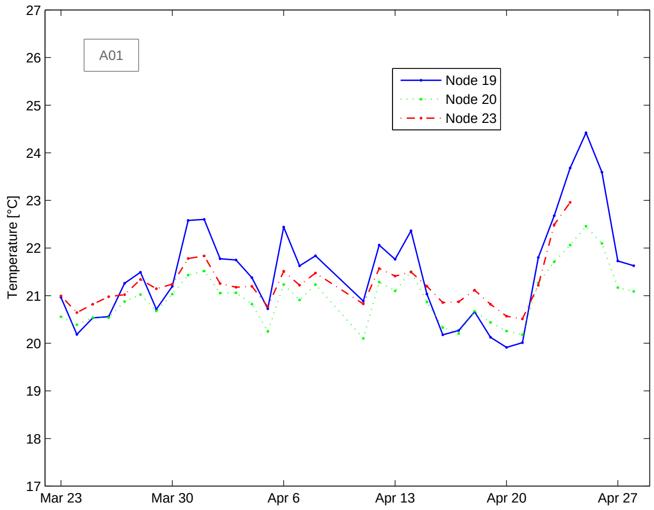



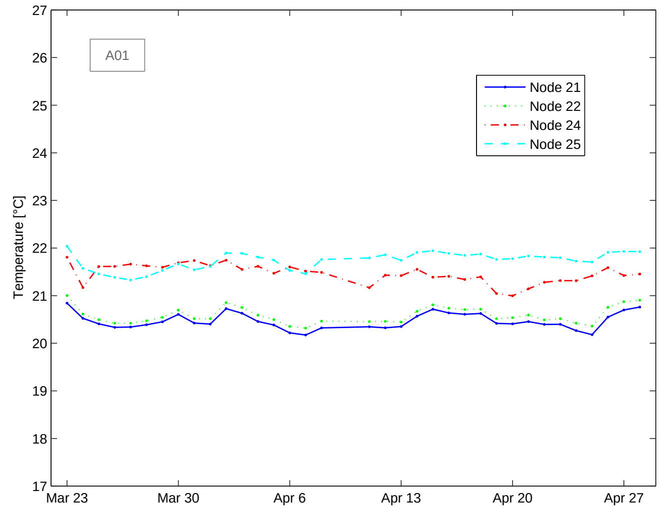



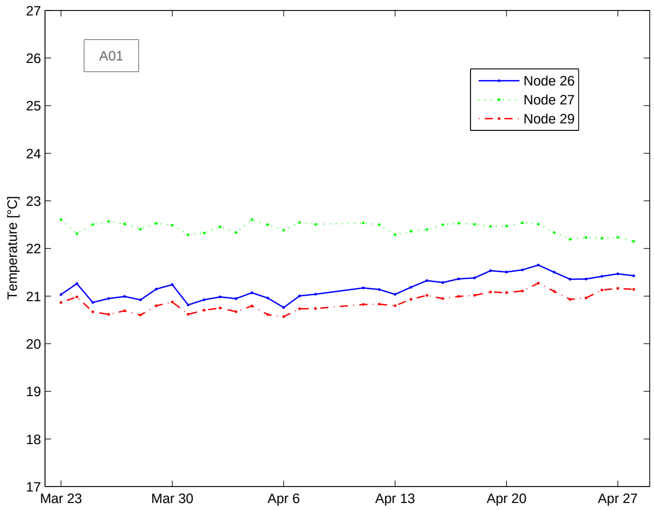



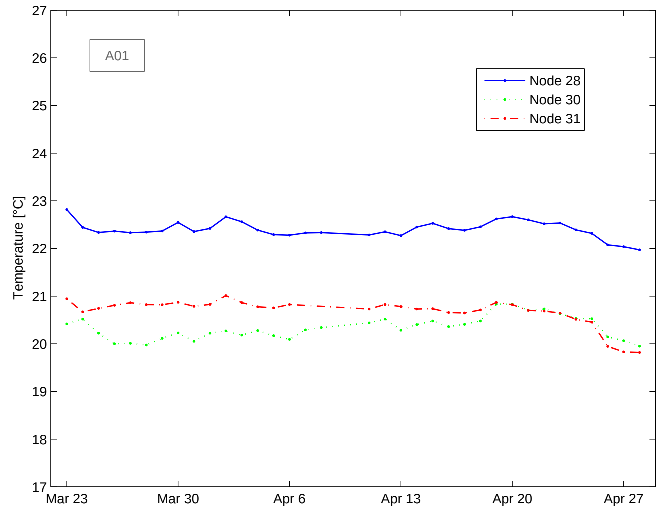



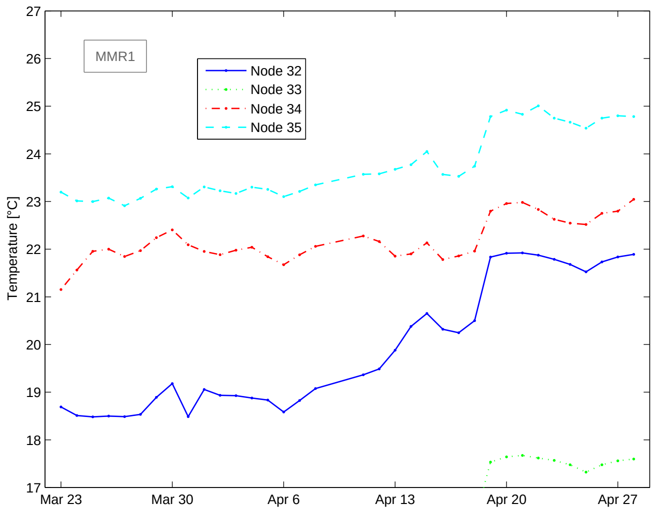



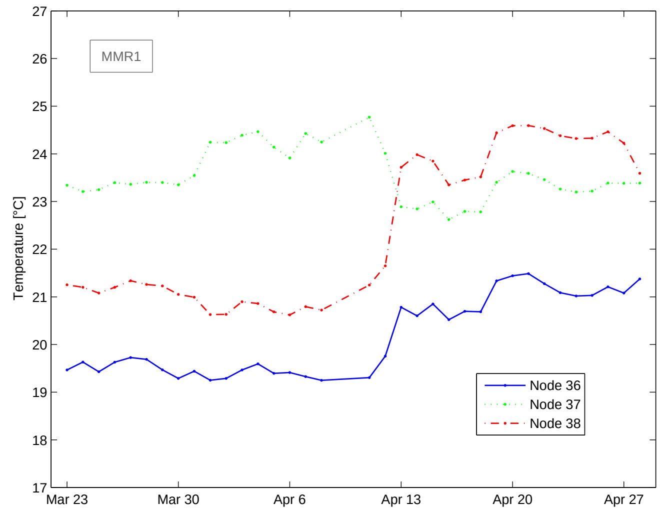

A01
Node 19
Node 20
Node 23
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

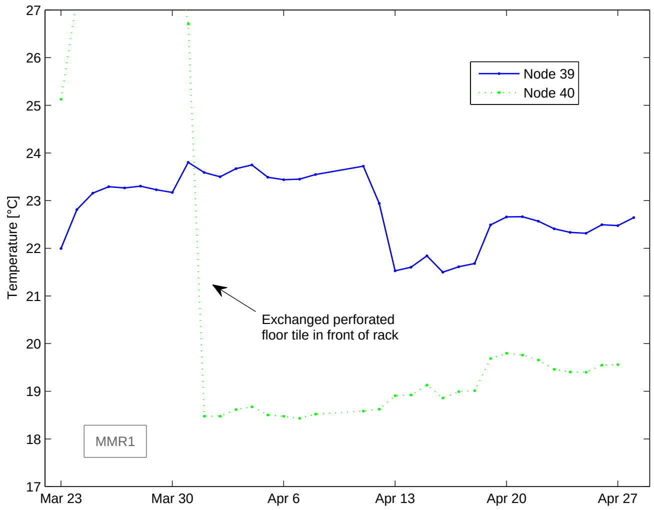

A01
Node 21
Node 22
Node 24
Node 25
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

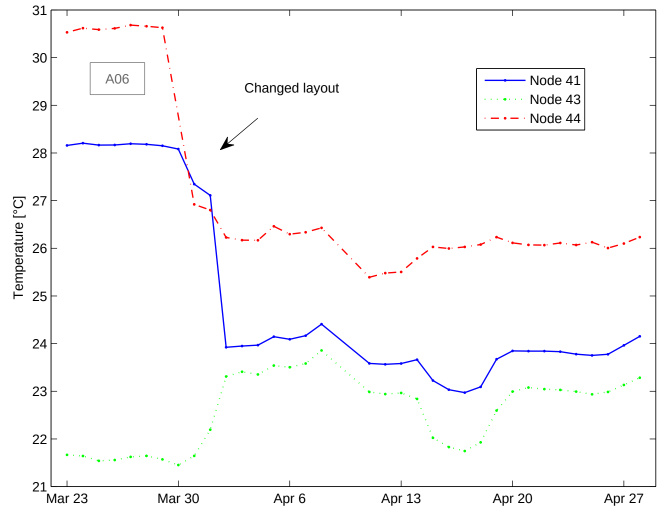

A01
Node 26
Node 27
Node 29
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

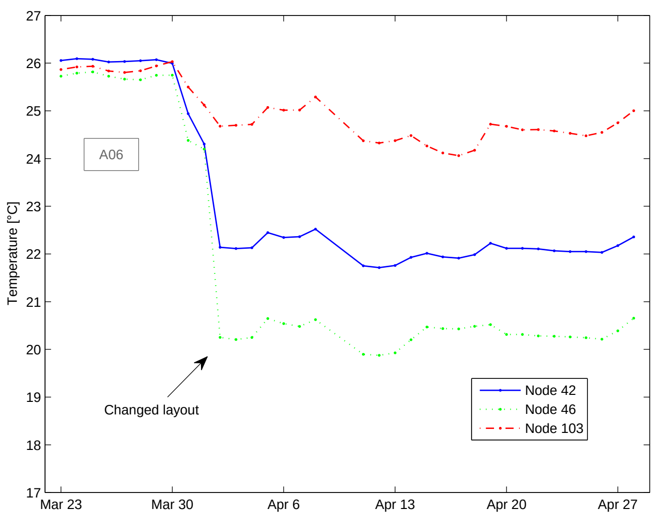

A01
Node 28
Node 30
Node 31
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

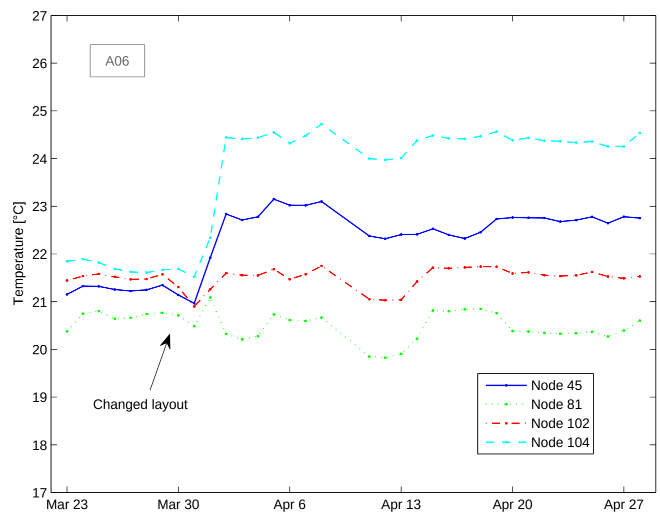

MMR1
Node 32
Node 33
Node 34
Node 35
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

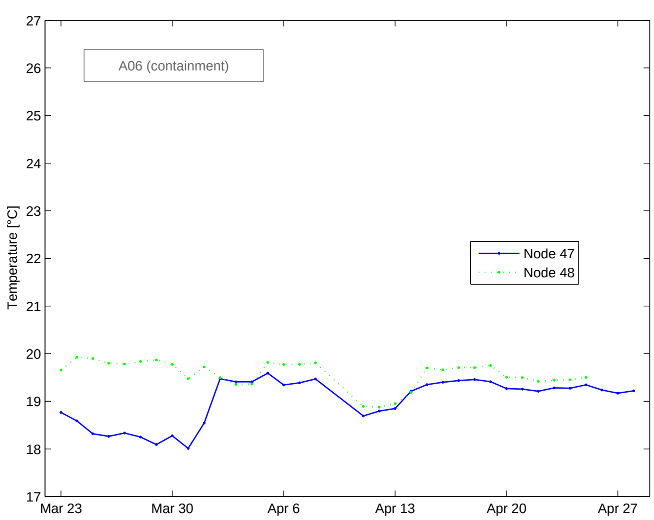

MMR1
Node 36
Node 37
Node 38
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

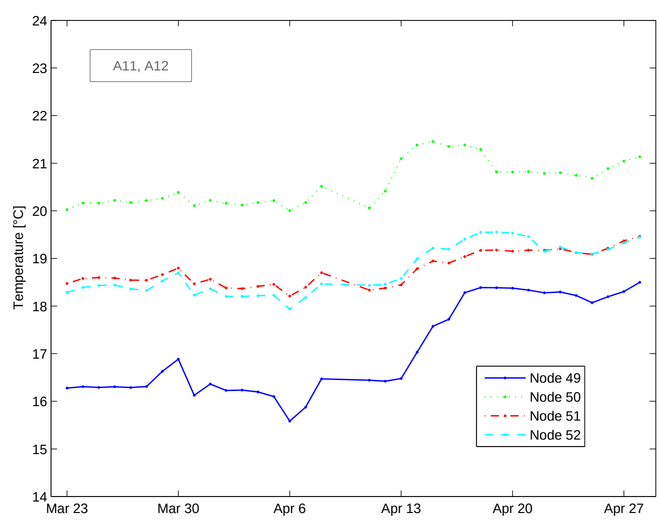

Node 39
Node 40
Exchanged perforated
floor tile in front of rack
MMR1
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

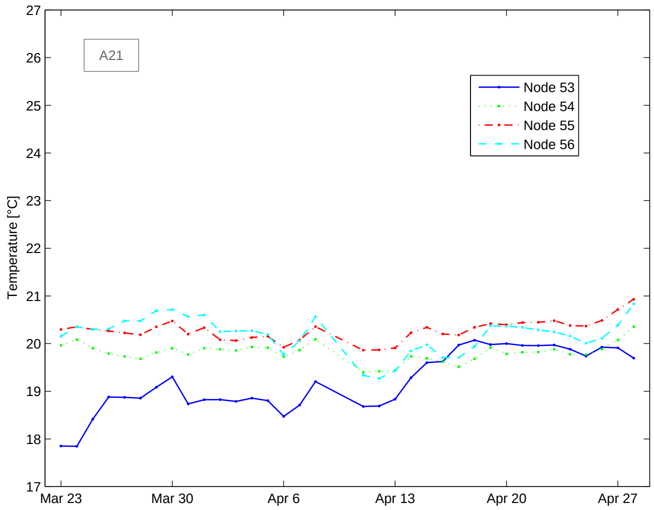

A06
Changed layout
Node 41
Node 43
Node 44
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

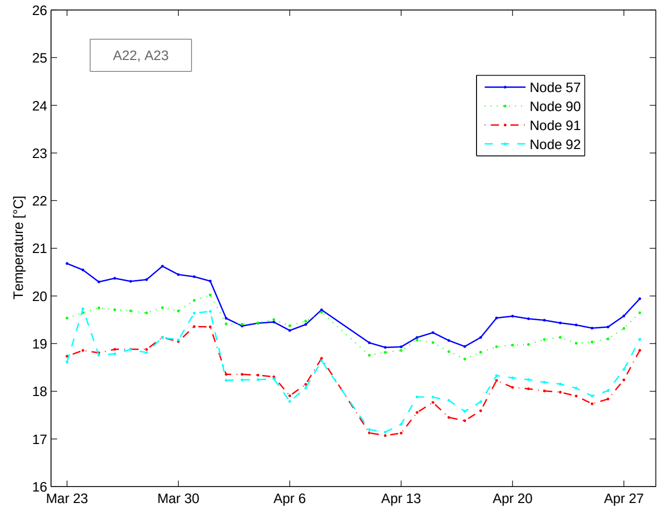

A06
Changed layout
Node 42
Node 46
Node 103
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

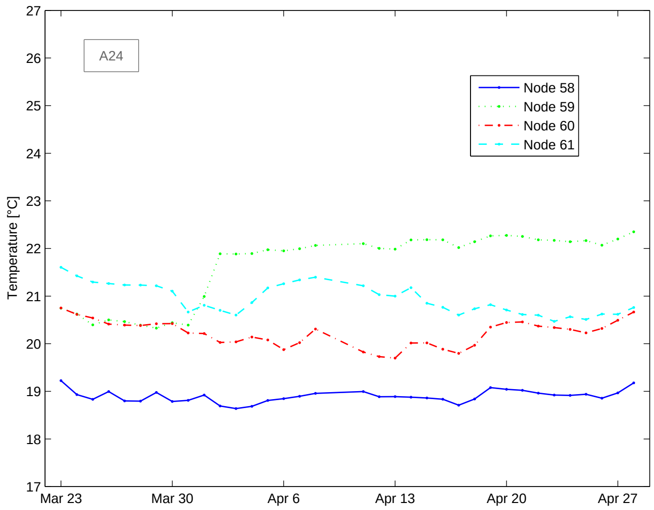

A06
Changed layout
Node 45
Node 81
Node 102
Node 104
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

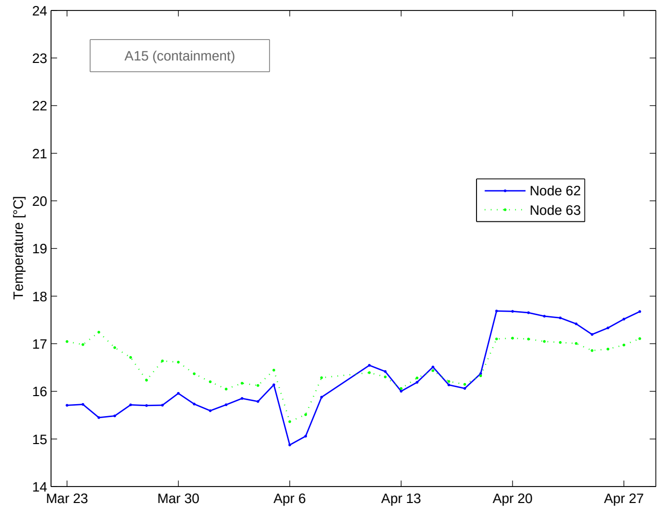

A06 (containment)
Node 47
Node 48
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

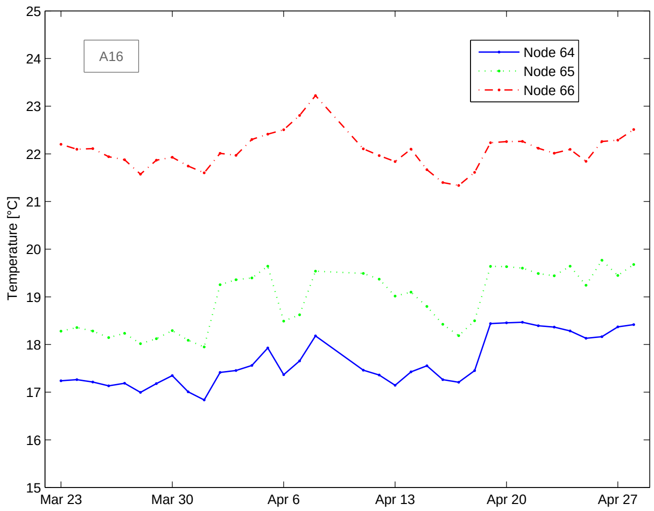

A11, A12
Node 49
Node 50
Node 51
Node 52
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

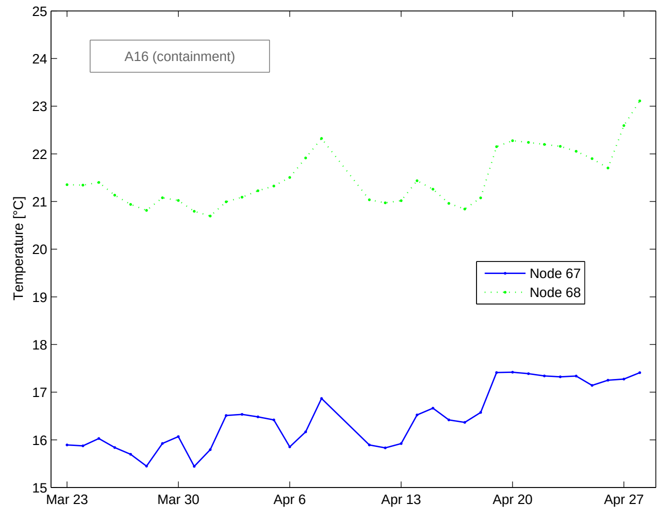

A21
Node 53
Node 54
Node 55
Node 56
Temperature [°C]
17 18 19 20 21 22 23 24 25 26 27
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

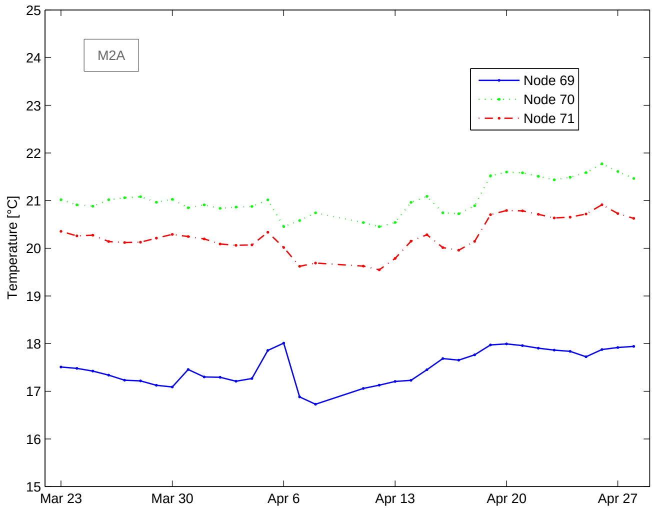

A22, A23
Node 57
Node 90
Node 91
Node 92
Temperature [°C]
16 17 18 19 20 21 22 23 24 25 26
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

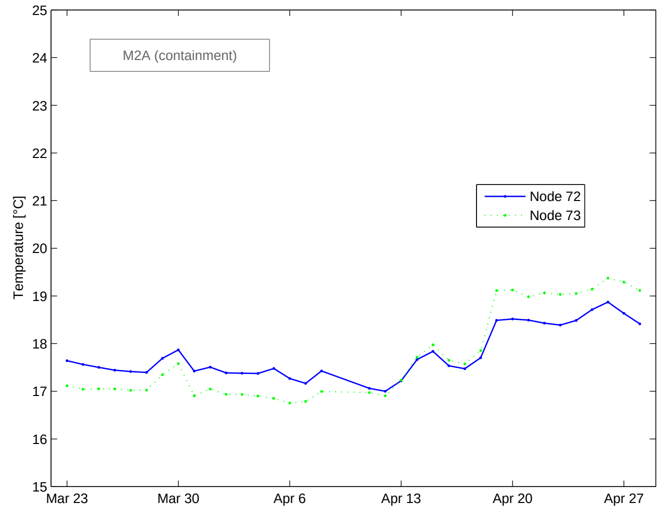

A24
Node 58
Node 59
Node 60
Node 61
Temperature [°C]
17 18 19 20 21 22 23 24 25 26 27
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

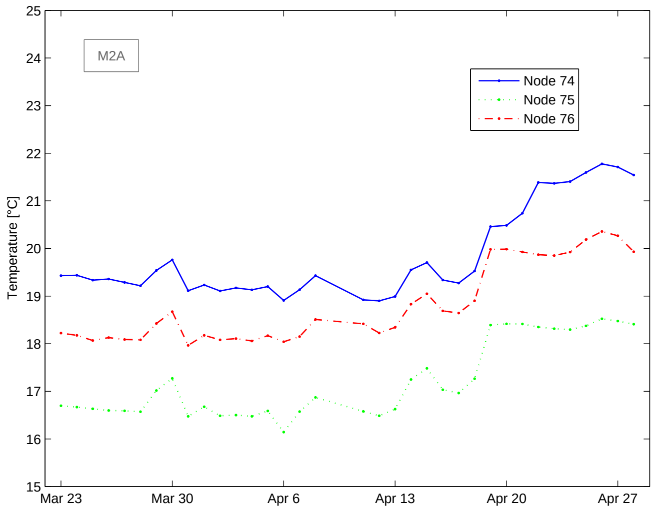

A15 (containment)
Node 62
Node 63
Temperature [°C]
14 15 16 17 18 19 20 21 22 23 24
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

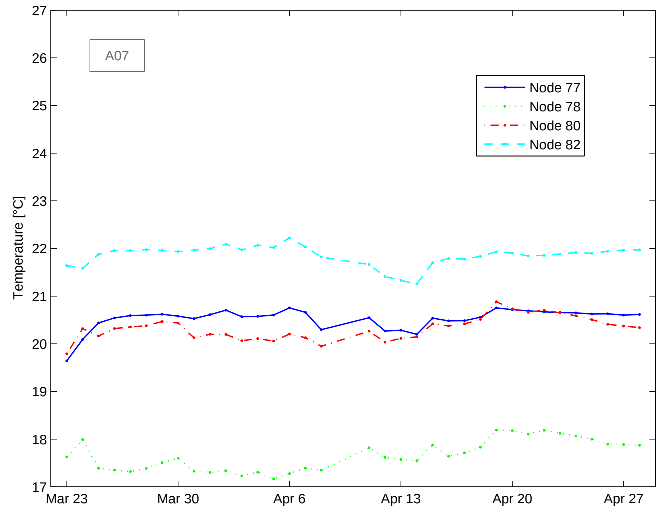

A16
Node 64
Node 65
Node 66
Temperature [°C]
15 16 17 18 19 20 21 22 23 24 25
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

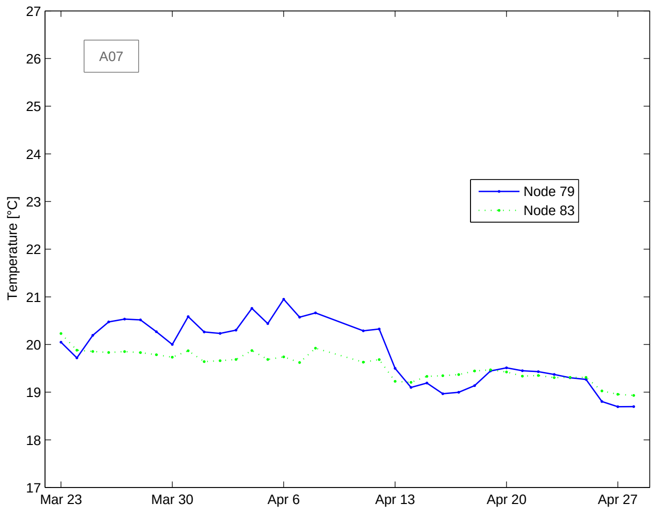

A16 (containment)
Node 67
Node 68
Temperature [°C]
15 16 17 18 19 20 21 22 23 24 25
Mar 23 Mar 30 Apr 6 Apr 13 Apr 20 Apr 27

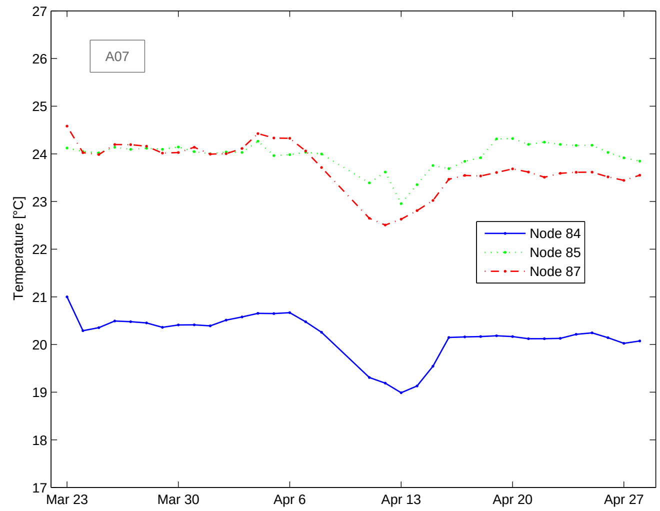

M2A
Node 69
Node 70
Node 71
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

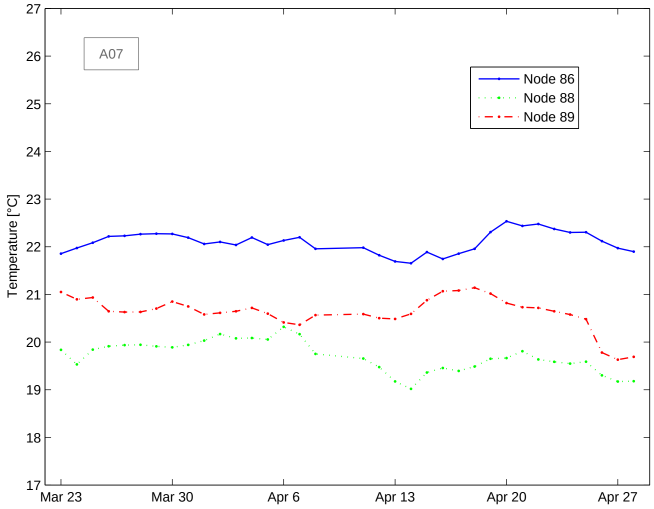

M2A (containment)
Node 72
Node 73
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

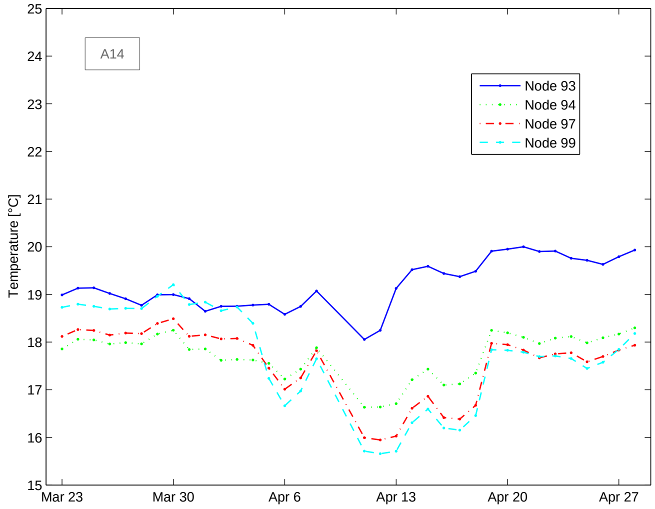

M2A
Node 74
Node 75
Node 76
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27

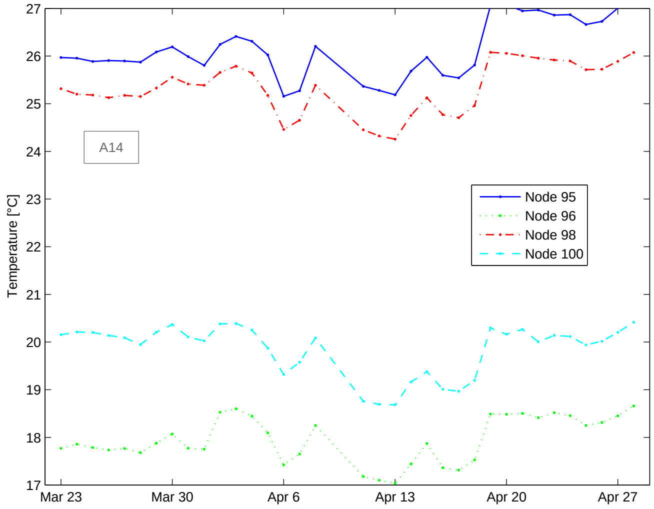

A07
Node 77
Node 78
Node 80
Node 82
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27


A07
Node 79
Node 83
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27


A07
Node 84
Node 85
Node 87
Temperature [°C]
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27


A07
Node 86
Node 88
Node 89
Temperature [°C]
27
26
25
24
23
22
21
20
19
18
17
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27


A14
Node 93
Node 94
Node 97
Node 99
Temperature [°C]
25
24
23
22
21
20
19
18
17
16
15
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27


A14
Node 95
Node 96
Node 98
Node 100
Temperature [°C]
27
26
25
24
23
22
21
20
19
18
17
Mar 23
Mar 30
Apr 6
Apr 13
Apr 20
Apr 27