Data Acquisition System of the CLOUD Experiment at CERN

Stefan K. Weber, Giovanna Lehmann Miotto, João Almeida, Pascal Herve Blanc, António Dias, Giulio Malaguti, Hanna E. Manninen, Joschka Pfeifer, Sylvain Ravat, Antti Onnela, Serge Mathot, Jasper Kirkby, António Tomé, and António Amorim

***Abstract*—The Cosmics Leaving OUtdoor Droplets (CLOUD) experiment at the European Organization for Nuclear Research (CERN) is investigating the nucleation and growth of aerosol particles under atmospheric conditions and their activation into cloud droplets. The experiment comprises an ultraclean 26 m^3 chamber and its associated systems (the CLOUD facility) together with a suite of around 50 advanced instruments attached to the chamber via sampling probes to analyze its contents. The set of instruments changes for each experimental campaign according to the scientific goals. The central function of the CLOUD DAQ (data acquisition) system is to combine the data from these autonomous and inhomogeneous instruments into a single, integrated CLOUD experiment database. The DAQ system needs to be highly adaptable to allow a fast setup over a single installation week at the start of each campaign when the instruments are brought to CERN and installed at the CLOUD chamber. Each campaign requires high flexibility and fast response to changes in instrument configuration or experimental parameters. The experiments require online monitoring of the physical and chemical measurements with delays of only a few seconds. In addition, the raw data, the monitoring databases, and the processed data must be archived and provided to the international collaboration for both real-time and later analyses. We will describe the various components of the CLOUD DAQ and computing infrastructure, together with the reasons for the chosen solutions.**

***Index Terms*—Centralized control, data acquisition (DAQ), data handling, data processing, data storage systems, industrial control, monitoring, remote monitoring.**

I. Introduction

THE Cosmics Leaving OUtdoor Droplets (CLOUD) experiment at the European Organization for Nuclear Research (CERN) studies the formation and growth of aerosol particles and their impact on clouds [1]. This is carried out under controlled atmospheric conditions, with precise control of the chamber characteristics such as temperature, humidity, or light exposure. A detailed knowledge of various trace gas concentrations, e.g., sulfuric or nitric acid from mass spectrometers, is also required. Such controlled measurements are performed during campaigns, occurring once or twice a year with an approximate duration of three months, during which data are collected 24 h/day and seven days per week. During these measurement periods, researchers from collaborating institutes operate their instruments at the CLOUD chamber, to obtain a comprehensive understanding of the processes involved. The CLOUD collaboration has conducted 16 campaigns at CERN so far.

At the start of each campaign, around 50 instruments are brought to CERN and attached to the CLOUD chamber via sampling probes. At the end of each campaign, the instruments are returned to the institutes for deployment in field and aircraft observations. To maximize experimental measurements, a fast integration at CLOUD is therefore vital. Moreover, as the campaign proceeds, flexibility, fast response, and immediate interpretation of the data are key to choosing the optimum operating conditions for the experiments. This makes quasi-real-time online monitoring of the experimental data mandatory. Furthermore, the raw and processed data from each campaign must be stored and made available to the collaboration. This includes the data from both the instruments and the CLOUD facility systems (gas, illumination, thermal, chamber mixing fans, high voltage, and beam counters). This ultimately allows interpretation and analysis of physical and chemical phenomena that take place in the chamber.

The data are collected from individual instrument files that are continuously updated by the autonomous data collection system of each instrument. This requires frequent flushing of data buffers to raw data files for each individual instrument. Our approach ensures that data are recoverable in the event of any central DAQ downtime while maintaining minimum modification of the instrument DAQ system, which must operate both at CLOUD and in the field. The CLOUD DAQ needs to handle instrument data files that may be open for widely different time intervals ranging from seconds to an entire campaign of roughly nine weeks. The instruments operate on a wide range of platforms, including numerous Linux (https://www.linuxfoundation.org) distributions (RPM-based and Debian-based), and various Microsoft Windows versions (ranging from Windows XP

Manuscript received June 14, 2020; accepted August 24, 2020. Date of publication September 15, 2020; date of current version November 20, 2020. This work was supported in part by the Marie Skłodowska-Curie Innovative Training Network Fellowship of the European Commission's Horizon 2020 Program under Contract 764991 CLOUD-MOTION and in part by the Portuguese Science and Technology Foundation (FCT) under Project CERN/FIS-COM/0014/2017. The Associate Editor coordinating the review process was Daniel Belega. *(Corresponding author: Stefan K. Weber.)*

Stefan K. Weber, Giovanna Lehmann Miotto, João Almeida, Pascal Herve Blanc, Giulio Malaguti, Hanna E. Manninen, Joschka Pfeifer, Sylvain Ravat, Antti Onnela, and Serge Mathot are with CERN, CH-1211 Geneva, Switzerland (e-mail: stefan.karl.weber@cern.ch).

António Dias and António Amorim are with CENTRA, Universidade de Lisboa, 1849–016 Lisbon, Portugal, and also with the Faculdade de Ciencias, Universidade de Lisboa, 1849–016 Lisbon, Portugal.

Jasper Kirkby is with CERN, CH-1211 Geneva, Switzerland, and also with the Institute for Atmospheric and Environmental Sciences, Goethe University Frankfurt, 60438 Frankfurt, Germany.

António Tomé is with the Institute Infante Dom Luíz, University of Beira Interior, 6201–001 Covilhã, Portugal.

Digital Object Identifier 10.1109/TIM.2020.3023210

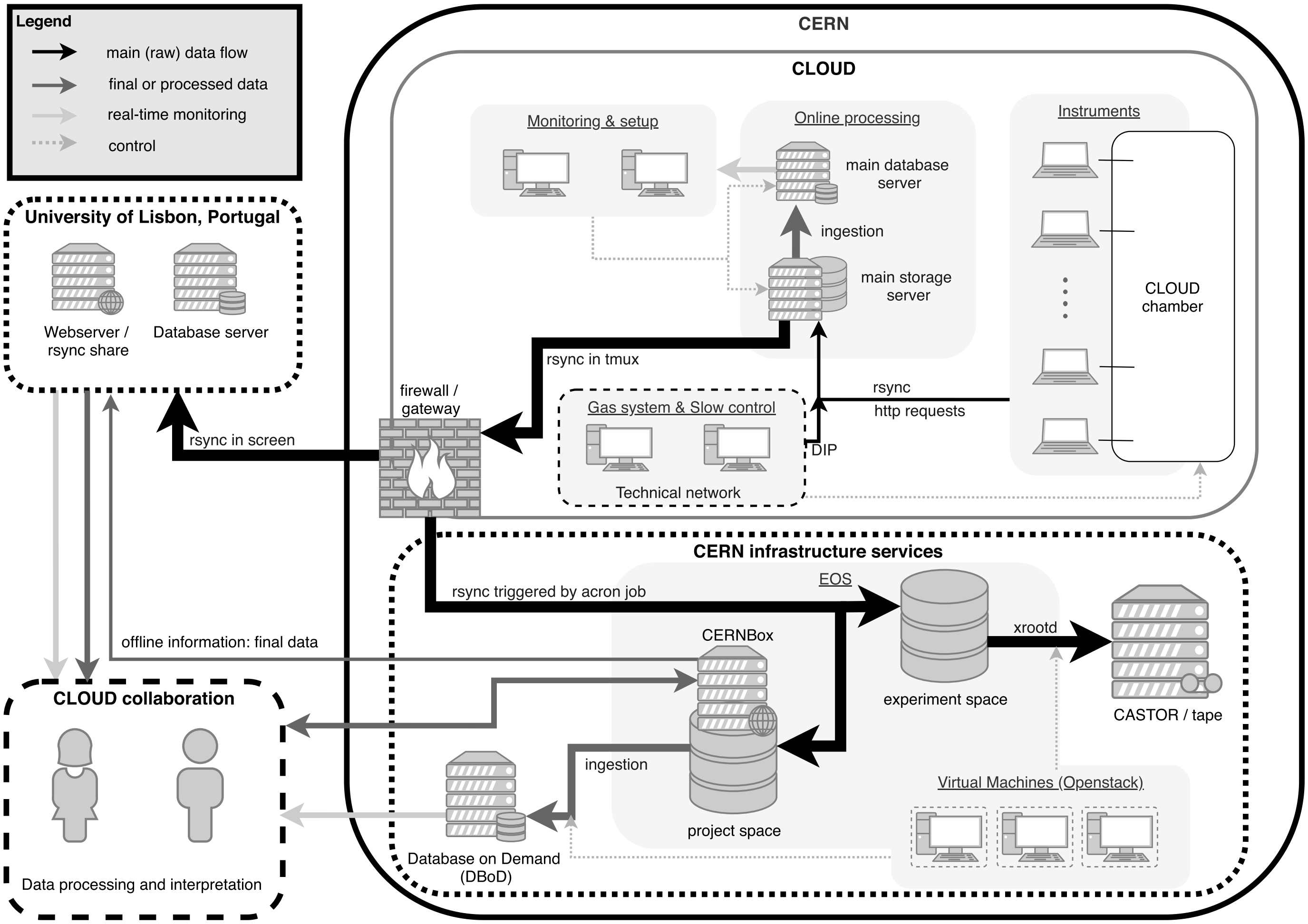

Fig. 1. Overview of the DAQ structure of the CLOUD experiment. The data acquisition and processing can be divided into four main information blocks: local CLOUD DAQ, CERN infrastructure, remote Lisbon server, and the CLOUD researchers. The latter request and obtained data from the servers and (for instrument experts), also directly from the autonomous instrument computers (via external storage devices). The different data and information flows for the main (raw) data flow, the final or processed data, the real-time monitoring, and the control are depicted using different arrow styles.

to Windows 10, https://www.microsoft.com). Some of the instruments are also based on real-time operating systems. In addition, a variety of different frameworks for readout, storage, and the respective data analysis procedures are encountered. These are based on an equally wide range of platforms, such as standard open-source solutions (including C/C++ (https://isocpp.org), python (https://www.python.org), R (https://www.r-project.org), julia (https://julialang.org), JavaScript, HTML (https://html.spec.whatwg.org/multipage), CSS (https://www.w3.org/TR/CSS/#css), ShellScript, MySQL (https://www.mysql.com), Qt (https://www.qt.io), and Cygwin (https://www.cygwin.com)] and also closed-source and proprietary software [including MATLAB (https://www.mathworks.com), LabView (https://www.ni.com), Igor (https://www.wavemetrics.com), and TOFWERK software (https://www.tofwerk.com)]. A further challenge is that the instrument DAQ systems have frequently been developed by companies or individuals who are unavailable to make modifications to the instrument software or data format.

The aim of this article is to describe how the CLOUD data acquisition system combines diverse standalone acquisition systems into a single, integrated DAQ system. This acquisition system is optimized for immediate data display and easy database replication on independent sites. We present our solutions to these challenging design requirements in Section II, together with an overview of the CLOUD DAQ. Section III focuses on the methods we use for the data acquisition (see Section III-A), online monitoring (see Section III-B), backup and related final data analysis (see Section III-C), remote access and instrument control (see Section III-D), and slow control (see Section III-E). Section IV discusses the main issues observed during the last experimental run and suggests possible solutions. We summarize the main aspects of the CLOUD DAQ system in Section V.

II. General Computing Infrastructure

Fig. 1 shows the general infrastructure of the CLOUD experiment divided into four computing subsystems. The interplay of the various systems as well as their interface with the researcher is discussed in the following.

A. Network Layout

Since the experimental area is located at CERN, the main computing structure is within a subnetwork inside CERN's

General Purpose Network (GPN) [2]. The CLOUD experiment network, therefore, hosts the various (mainly external) instruments and data collection hardware. Since the CLOUD subnetwork is deliberately closed to prevent any interference with data taking, remote access is only granted to instrument experts for maintenance and configuration changes (see also Section III-D).

There is an additional network layer, called the technical network (TN) [2] used for instrument control, readout, and communication of systems permanently at CERN. These are the so-called slow control systems mainly for operation and readout of the various CLOUD chamber systems. The slow control comprises the entire gas system as well as numerous other instruments, including the various light sources and light detection systems. These are controlled via standard CERN software solutions and infrastructure operated on CERN's TN (see also Section III-E).

From a CLOUD system administrator's point of view, the CLOUD subnetwork and access to the TN is handled via standard CERN IT infrastructure. Access between the various network layers is internally handled via firewall rules assigned to numerous sets that, in turn, comprise the machines requiring specific rights.

The main storage server is exposed to the TN to also retrieve the raw data originating from the slow control part of the DAQ. The Linux gateway machines are defined and exposed to the GPN to allow data transfer from the closed CLOUD network. To facilitate remote access to the graphical user interface (GUI) of instrument computers (running Windows), standard CERN IT gateway servers are trusted by the CLOUD experiment network.

B. Data Monitoring, Access, and Analysis

Since the CLOUD network is closed by design, additional and independent monitoring instances are required.

The so-called first storage instance comprises the centrally collected raw data from all (roughly 50) instruments used to simulate, control, and ultimately study the atmosphere artificially created in the CLOUD chamber. This first storage instance is also used to build the first instance of the monitoring database, providing almost immediate monitoring capability (with an approximate maximum latency of 10 s) within the CLOUD network. We also compute and include (clearly labeled) preprocessed data in real time, which are also ingested to database and transferred with the raw data. For simplicity, "raw data" refers to both the actual instrument raw data as well as preprocessed monitoring data.

While the second storage instance is based on the backup of raw data (which also increases the latency to approximately 5 min[1]), independent monitoring instances are provided from two sites, the University of Lisbon and CERN. This second instance is set up as a global service for the entire CLOUD collaboration and furthermore provides high availability combined with redundancy in monitoring as well as access to raw data.

[1]The bottleneck here is the transfer speed of raw data to the different locations.

The final analysis of the data is typically done after the measurement campaigns and relies on independent off-line infrastructures. At Lisbon, a desired simplified user management leads to a single read-only account for the whole collaboration that is used for both access to raw data and the databases. Access to data served by the databases is provided via retrieval applications. The database stores raw data, monitoring data, and analyzed data. The raw data on the Lisbon server are accessible via a simple web interface. At CERN, each CLOUD user is required to use their personal CERN computing account, allowing for fine-grained access control. The raw data on CERN storage is distributed using CERNBox (http://cernbox.web.cern.ch), which provides direct access to the underlying EOS storage space via a web browser or specific clients [3], [4]. In case of access to CERN databases, port forwarding is facilitated by the access applications.

Final processed data, analyzed offline by the instrument experts, are centrally collected and distributed via CERNBox which is also mirrored to the Lisbon site twice a day. Since numerous final analyses depend on data from several instruments, it is advantageous to access the ingested data via the database in order to get an overview or retrieve raw data in a common format.

C. CLOUD's Computing Hardware

The CLOUD data collection infrastructure at CERN is based on standard CERN IT-deployed server nodes, equipped with two 1 TB SSDs,[2] 20 physical CPUs (Intel Xeon CPU E52630 v4, 2.20GHz), and 128 GB RAM. Furthermore, the nodes support up to 10 GbE.[3] The abovementioned first storage instance is a so-called Just a Bunch Of Disks (JBOD) spanning a redundant array of independent disks (RAID) instance[4] [5]. This JBOD is attached to CLOUD's main collection and processing server and consists of 24 HDDs each with 6-TB storage capacity.

III. Architecture and Methods

In this section, the methods and the implemented architecture are introduced. Throughout this section, the interplay as well as the role of various parts in Fig. 2 are explained in greater detail.

A. Data Acquisition and Timing

The main storage server is the central collection point of raw data gathered from the individual DAQ system of each instrument. The core of the raw data synchronization procedure is solely based on the standard rsync tool and protocol (https://rsync.samba.org/). The main server is running a rsync daemon to which the individual clients

[2]An RAID1 is spanned over both SSDs to guard against disk failure.

[3]10 Gigabit Ethernet.

[4]The JBOD is connected to a server via a Serial Attached SCSI with SCSI denoting Small Computer System Interface (SAS) cable supporting transfer speeds up to 12 Gbit/s. To maximize data availability on the JBOD and guard against disk failure, a software RAID10 is implemented using the mdadm package (http://neil.brown.name/blog/mdadm and https://git.kernel.org/pub/scm/utils/mdadm/mdadm.git/).

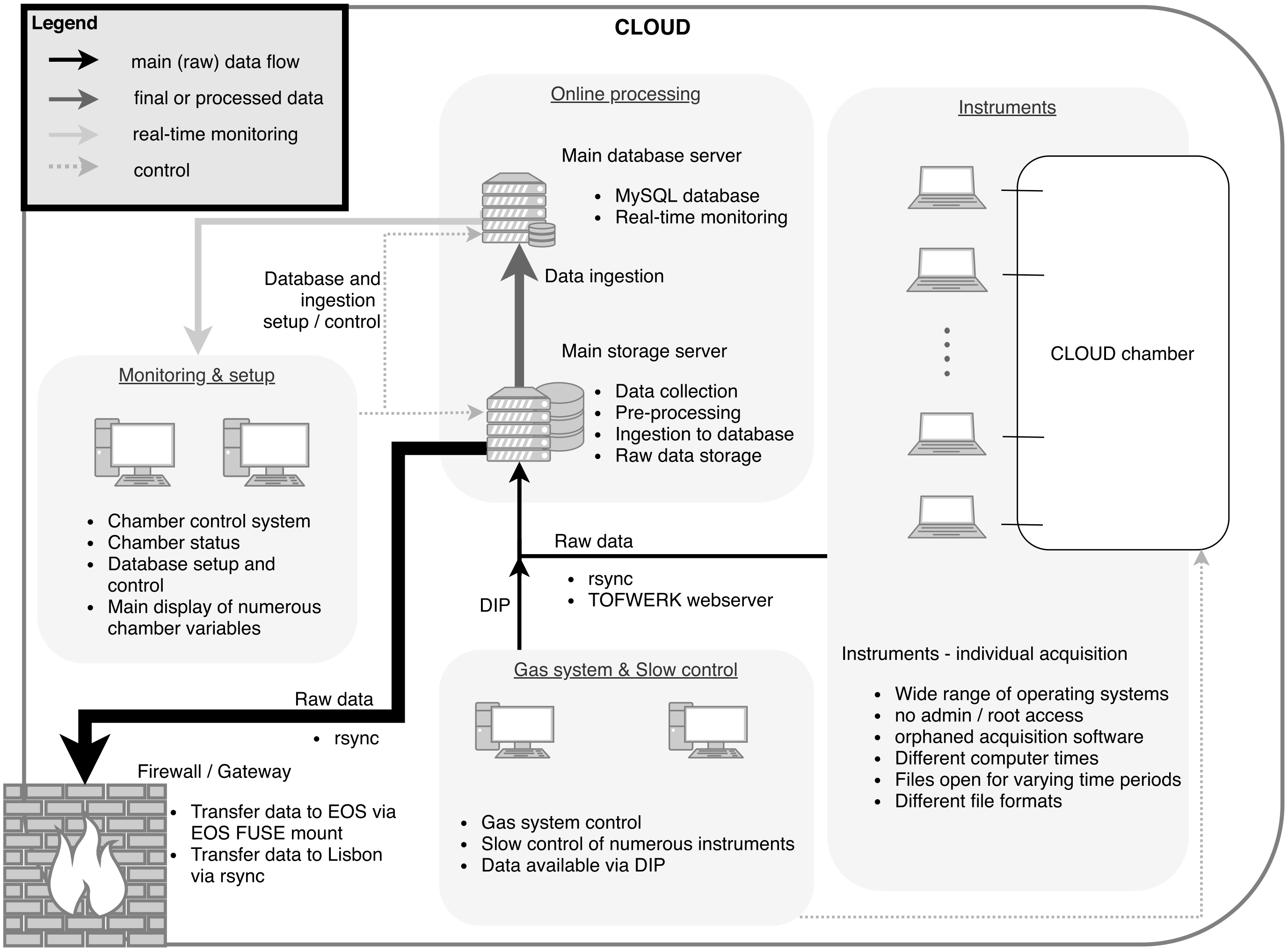

Fig. 2. Data flow within the CLOUD network. The main elements are data acquisition (see Section III-A), online monitoring (see Section III-B), and slow control (see Section III-E). Data are centrally collected from the various external instruments as well as the slow control systems by the main server and ingested to a database for monitoring. Data are further duplicated to external storage systems via the gateways.

connect. In order to ensure permanent synchronization, the clients are required to call rsync repeatedly, e.g., in a loop. If the client has a Linux operating system, the rsync script is run repeatedly in short intervals (via a GNU screen (https://www.gnu.org/software/screen) or tmux (https://github.com/tmux/tmux/wiki) session) to ensure that data are synchronized permanently. If the client, as in most cases, has a Windows operating system, the same procedure is used after porting rsync to Windows with cygwin.

In comparison with a data collection agent for each instrument, we found that a simple script invoking rsync in a loop is more effective in providing direct debugging and access to error messages. This ultimately speeds up and simplifies the identification of the source of possible problems that may arise. There are several advantages in using rsync as the main data transfer routine. In particular, its algorithm allows to identify and transfer only the file changes [6] in a transparent way using the so-called delta-transfer algorithm. This automatically reduces network traffic and makes fast and immediate transfers possible.

1) Instrument Requirements: Most of the instrument data acquisition applications write the data to ASCII files that are frequently appended. This generally involves a local file system where the append statements also flush the buffers to file.

There is, however, one important class of instruments that generate binary files that cannot be accessed unless it has been completely written and explicitly closed by the instrument acquisition software. Binary files, in particular h5 files, are produced by the time-of-flight (ToF) mass spectrometers [7]. For these instruments, another low-latency solution is needed to access the data. Furthermore, one needs to avoid repeatedly calling rsync to update files that are only meaningful after being fully written, i.e., properly closed by the instrument acquisition software. The latter is handled by excluding all files from synchronization that has been modified recently, e.g., within the previous 120 s. To avoid binary h5 files for low-latency monitoring, the solution has been provided by the company TOFWERK AG (https://www.tofwerk.com), which manufactures the instruments and provides the acquisition software. This solution, in addition to the standard

TOF acquisition to binary files, acts as a web server and provides remote data access at the instrument level. This allows a parallel data flow, implemented completely independent of the binary data files, via json-based network communication. Using this service, one gains access to several low-level quantities of the mass spectrometer, including the raw data. Data are usually extracted every second. Since this method allows to access the "instantaneous" counting rate per channel during the last measurement cycle,[5] the signal-to-noise ratio needs to be improved. Thus, a moving average (typically over 300 s) is applied to the data. This mechanism has been found suitable for online display, monitoring, and immediate interpretation.

In order to optimize the tradeoff between precision and acquisition speed, we collect one data point per second for a specified mass range in each ToF. This provides adequate online monitoring by summing the counts in the specified mass boundaries provided by the instrument experts. While in principle it is possible to query the mass range more often, a 1-Hz data frequency is sufficient for a fast interpretation of the data. For the offline analysis, the binary files are used. They gather data for longer time periods and provide improved signal-to-noise.

Another class of instruments, namely, the slow control systems, are also treated in a special way since they are operated on the TN and centrally managed by the EP-DT department at CERN (see Section III-E). Besides being configured to archive data independently, the data are also published via CERN's Data Interchange Protocol (DIP) [8], [9]. The architecture of the application to collect this data on-the-fly includes the implementation of dedicated processes subscribing to the DIP publications on the TN and appends the data continuously to files. The resulting files are processed and the data are ingested to the databases following the steps described in the following (see Section III-B).

Most of the instruments require further preprocessing to make the data suitable for online display and interpretation. Moreover, certain physics quantities that are needed for online monitoring to guide the experiments—such as particle nucleation rates—are calculated by combining the information from several instruments. These processes are also carried out on the main server, where they run repeatedly and append their results to various files that are then ingested into the database. The CLOUD data acquisition system described earlier immediately transmits the newly gathered data from all CLOUD instruments onto the main data collection server within a maximum delay of around 10 s.

2) Drifts in Time: An important problem when dealing with numerous independent instruments whose data are only connected via their time stamp is the synchronization of multiple clocks. In principle, all computer clocks can be synchronized to Universal Time Coordinated (UTC; the CLOUD standard) to millisecond precision simply by polling Network Time Protocol (NTP) servers [10]–[12], whereas CLOUD requires only a precision of about 1 s. However, in practice, the instrument clocks can drift from UTC by 100 s or more during a ten-week

[5]Typically, a measurement cycle is around 25 ns long.

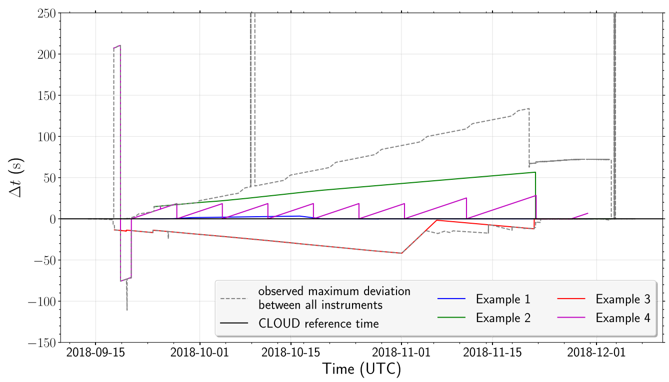

Fig. 3. Measured time deviations during a single campaign for several example instrument computers on the CLOUD network. The maximum deviation, considering all instruments, is also displayed. The CLOUD reference time is continually synchronized to UTC with standard CERN NTP servers. This shows the need for time synchronization and monitoring of time drifts during data collection. The time drifts are monitored for each instrument and corrected offline.

campaign (see Fig. 3). There are several reasons for this. In some cases, the default NTP polling interval is a reserved system option and is by default set to several days, which is too infrequent. Instrument computers are often set up with a user account lacking administrator privileges and therefore not allowing more frequent NTP polling. Furthermore, reboot of an instrument computer occasionally leaves an incorrect timezone configuration, leading to the most extreme deviations seen in Fig. 3. Time drifts are especially poor for instruments with badly designed National Instruments LabView software using its associated clock; here, we have found it necessary to restart all LabView applications frequently (every 24 h or less) to resynchronize the internal LabView clock with the computer clock. In short, although polling an NTP server is simple in principle, it is difficult in practice to ensure that excellent time synchronization is maintained for each one of around 50 autonomous computers that lack the central administrator's control. Our solution is twofold: we use an open-source Simple Network Time Protocol (SNTP) [13] tool called NetTime (http://www.timesynctool.com) and we continually record the difference of the computer clock to UTC. These time corrections are centrally collected with the raw data and applied to the offline processed data.

B. Online Monitoring

The CLOUD event building for quasi-real-time online monitoring is performed when writing the data to the time-indexed database. A central MySQL database management system (DBMS) is running on a dedicated server solely hosting the monitoring database inside the CLOUD network. This database is exclusively for online monitoring in the CLOUD control room at CERN during data taking.

1) Database Design: The database schema design is normalized and follows the design guidelines.

1) Data from every instrument are assigned mainly to a single database table that is mapped by the DBMS to a separate file on the host filesystem. The data write operations get exclusive locks for each instrument making it easy to ingest different instruments in parallel.

2) In most instruments, a collection of different data channels is read simultaneously. They are mapped to different columns (attributes) in the instrument table.
3) The whole table (entity) is indexed by the measurement time, stored as an 8-byte integer with the unix timestamp format expanded to milliseconds. The concept of the time interval (indexing also to the time of the next measurement) is avoided due to its impact on performance. While reading the data for a specific time or for a time series, the query is performed by decreasing the initial time value and increasing the final time value by a few seconds, guaranteeing that the relevant data updates are included. This has a negligible impact on the performance of retrieving time series.
4) In case the instrument generates a homogeneous and instantaneous array of data, the whole array is stored in a single data block that is transparent to the DBMS. The data are packed and unpacked by the CLOUD data access and monitoring applications.
5) Only a few instruments make data available as independent channel updates. For those, a small set of special tables are created (according to the data type) and indexed by both time and channel identification. The DAQ and monitoring applications automatically retrieve the time series of a single channel. Mandatory updates of all channels for an instrument are made available at regular timestamps, allowing the same time interval extraction mechanism to retrieve the required data as described earlier.
6) The database schema includes all metadata information, which guarantees the availability of all the information associated with the interpretation of data, data ingestion, and data processing. This further allows the full system to mainly run without the need for any extra configuration information.

Most of the concurrent operations involve querying time series data. By careful design of the database, the MyISAM storage engine proved to be suitable and very efficient for the task. The major part of write operations to database is due to the ingestion of raw data. A minor part also includes run table and logbook updates by users. Such write processes are, however, executed at a much lower rate than instrument data-related operations.

2) Ingestion: The write operations use specific database ingestion C++/Qt-based software, specifically developed within the collaboration. A single application is designed for flexibility and supports a huge diversity of text-based raw data files. In addition, numerous column addressing options are available, and simple formula parsing is supported at the ingestion level. For more complex processing, as often requested by instrument experts, preprocessing of raw data is still required. This is usually implemented by creating additional text-based files holding the preprocessed data and transferred alongside the raw data and therefore also available at the second monitoring instances.

The data ingestion at each database (CLOUD, CERN, and Lisbon) is managed by a single process management application that invokes a configurable number of parallel threads for the ingestion of each instrument's data. The database schema requires only some small bookkeeping tables for the different instruments. This implies that data ingestion for all instruments is fast and is easily achieved by running several ingestions in parallel. The local monitoring and database control structure is schematically shown in Fig. 4. Real-time monitoring is provided by fetching instrument data from the database and displaying it using a CLOUD-specific, C++/Qt-based application. The online monitoring and data analysis applications share most of their functionality. However, in the online monitoring version, an incremental fetch mechanism from database is implemented for the requested time series data to maximize the interface impression for display stability. In practice, we have found that a 30-s data refresh rate is optimal to update the CLOUD monitoring plots on display. To implement an effective load sharing between several data-intensive instruments, i.e., instruments producing large amounts of data, a limiting ingestion time is set for each instrument per ingestion cycle. In case an instrument tries to write longer than the set ingestion time, the write cycle for this instrument is stopped and the thread is allocated by another instrument ingestion process, again with a maximum duration of the set ingestion time for the current instrument write cycle. The processed data in each ingestion cycle are bookkept by writing the stopping point of ingestion to a text file. This includes the path to the datafile as well as the last time stamp of ingestion in this file. Lock files are used to flag the currently processed instrument data, thereby preventing several threads from working on the same instrument. To monitor the most recent data, the maximum ingestion time is set to a few seconds, enabling the writing of newly recorded data for all instruments. The ingestion process running on a single core is schematically shown in Fig. 5.

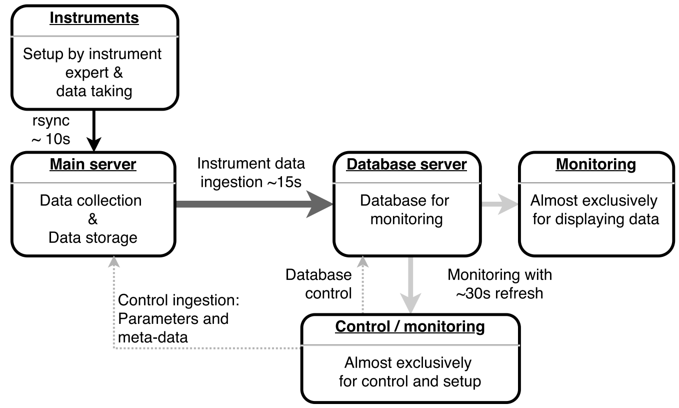

Fig. 4. Schematic of the online monitoring and database structure. The maximum time delays for the CLOUD data flow are indicated.

3) User Experience: More complex write operations are limited to run table and logbook entries. Those are done by the user approximately once every 15 min. While the logbook is used to comment on the data, the run table is needed to keep track of the run number, time, and experimental conditions. In CLOUD, the next integer run number is used for each major change of run conditions—a new experiment—whereas any transition within that experiment, such as a change of

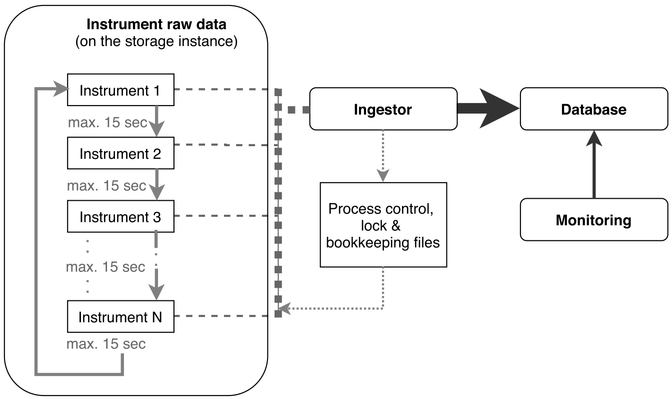

Fig. 5. Sketch of ingestion using a single process with a maximum ingestion time of 15 s. The ingestion is looping through instruments, one after another, and ingests data into the database for up to 15 s, i.e., the chosen ingestion time. In case all the data are ingested, it instantly moves on and picks the next instrument, for which the ingestion process is repeated. Once all instruments are handled, ingestion starts again from the first instrument. After each ingestion, the current status for the instrument is written to a file such that the next cycle for this instrument can immediately retrieve its starting point. In addition to bookkeeping mechanism and process id control checks, lock files indicate and ensure that only one ingestion per instrument is running as they are created at the start and deleted at the end of each write cycle. At CLOUD, there are usually 4–16 processes running in parallel.

gas concentration, is assigned the next stage number, which is appended as a decimal to the integer run number. Each CLOUD campaign is effectively a single time series for the whole 8–9 week period, which is structured via the run table. Although all experimental conditions are recorded along with the data from the instruments, the run table is essential in order to select the data sequence of interest for more detailed offline analysis and to be informed of the overall purpose and experimental conditions for each run and its stages.

To quickly identify potential problems with individual instruments, the current status of both the raw data synchronization and the current ingestion time is continuously displayed for each instrument. The former is done with the logging capability of the rsync daemon, following all available instruments with their last synchronization time. The latter is obtained from the last time a specific instrument was written to the database.

The same monitoring applications provide data monitoring for researchers who are not physically at CERN. Access to data is provided independently of two sites: CERN and Lisbon. In both cases, the database is locally rebuilt from raw data, transferred, and made available by the local services (see Section III-C). Rebuilding the databases from local raw data is easily achieved because the ingestion framework is designed to work from raw data, with all the necessary information for identical database setup stored in text files and transferred as part of the raw data. Since this data transfer involves some intermediate steps compared with data acquisition to the main server, the monitoring has a small delay (up to around 10 min) relative to the CLOUD control room.

C. Backup and Data Processing

As shown in Fig. 1, as soon as data are written to the main storage, they are further copied to different storage locations at CERN and Lisbon. Since CLOUD operates in its own subnetwork inside the CERN network, the data transfer must be made via gateway machines. These machines are part of the CLOUD network, though exposed to the GPN, and thus provide an interface between both. During campaigns, there is an additional data redundancy layer introduced at CLOUD, namely, an incremental tar based backup of the raw data, carried out once per day via a cron job.

In general, data transfer to the remote sites is done via rsync. Rsync synchronizes the raw data and all text-only database configuration metadata. The databases at Lisbon and CERN are rebuilt locally using the available metadata as well as the raw data. In this way, the SQL-based database mirroring processes is avoided. By verification, all the metadata to rebuild the databases are included in the files. This approach provides easy data distribution and an independent access to online monitoring databases while avoiding back-pressure to the central CLOUD data acquisition and monitoring infrastructure. The CLOUD scheme to provide both data and monitoring could readily be replicated for any other site worldwide. It is solely based on the rsync protocol for raw files, an independent data ingestion mechanism, and a control application for the external site.

Most of the replication and ingestion mechanism runs smoothly over the entire CLOUD campaign and only requires intervention if a database schema change has been executed at the CLOUD experiment. This has proven to be valuable since it confirms that the entire database can be easily rebuilt from raw data accompanied by the required metadata files.

Following each campaign, the raw data are carefully analyzed by instrument experts to extract the time series for each of the physical and chemical quantities required for offline data analysis. The time series for the processed data are uploaded to CERNBox and made available to all collaboration members. The data uploaded to CERNBox are mirrored to the Lisbon servers twice a day.

1) Processing at CERN: For the CERN backup, data are transferred via the gateway machines. Those are configured to receive acron jobs,[6] executing an rsync instance thta connects to the rsync daemon on the main server. This synchronizes the data from the first storage instance to the second storage instance at CERN, which is based on EOS [14], [15]. A Filesystem in Userspace (FUSE) (https://github.com/libfuse/libfuse) mount of EOS on the gateways avoids an additional rsync data transfer step. On the CLOUD side, the use of acron jobs is implemented to be process identification (pid) controlled via lock files. This ensures that only one synchronization instance is running at a time. Moreover, log files are written by both rsync as well as the pid controller to aid troubleshooting in case of problems. All log files are monitored via a log rotation.

Fig. 6 shows the CERN infrastructure used for monitoring, data distribution, collection, and backup. Data are transferred to EOS, more specifically to a so-called project space interfaced with CERNBox. This provides convenient

[6]Acron jobs are similar to cron jobs but allow for Kerberos authenticated jobs running on a specific host in the CERN network. Kerberos (http://www.kerberos.org) is the user authentication protocol used at CERN.

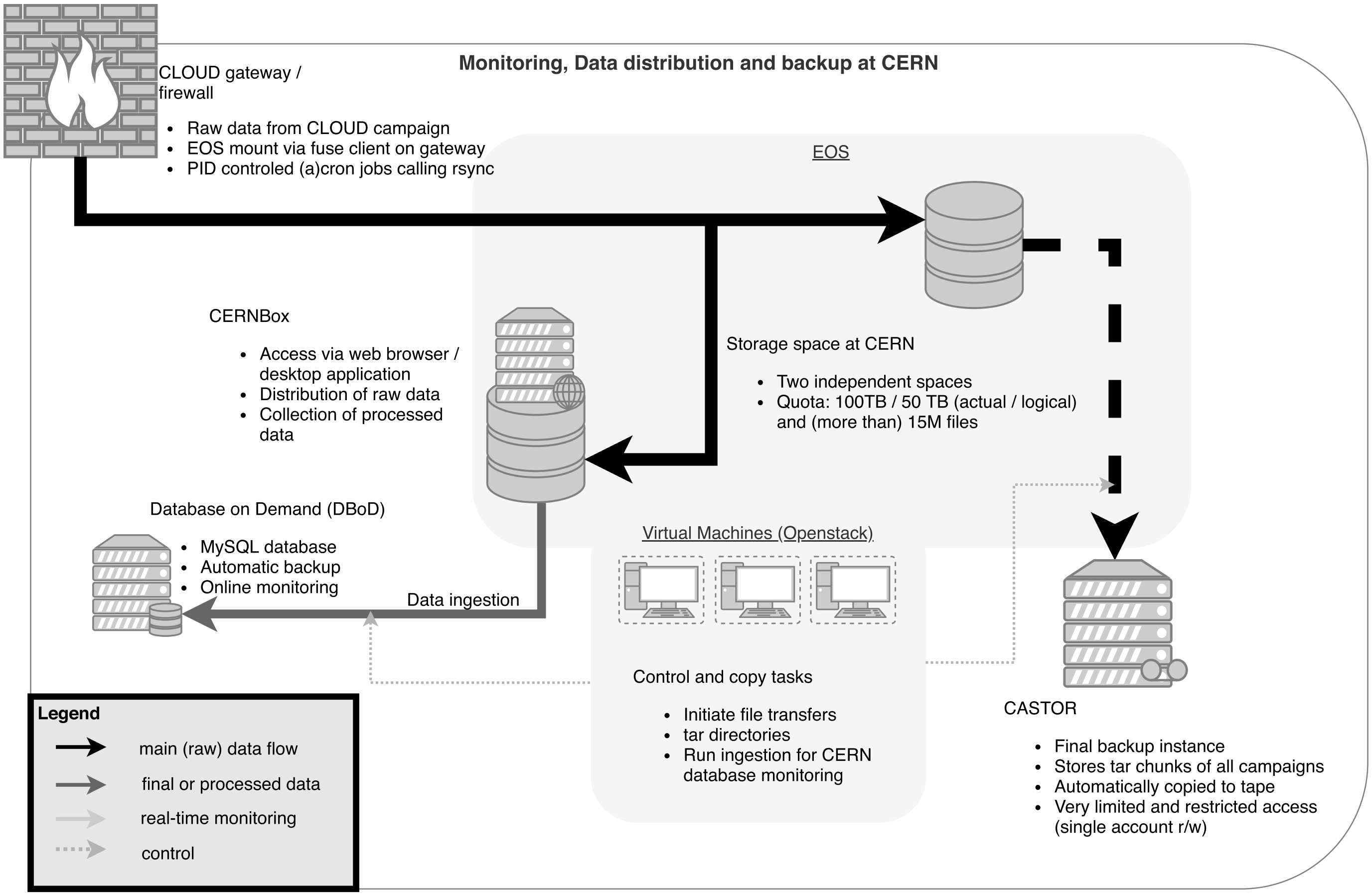

Fig. 6. Backup structure at CERN, within the CERN network. The backup as well as data distribution use standard CERN infrastructure.

access to raw and processed data via a web browser or desktop client. Another common way to access the data on the underlying EOS storage space directly from command line is via the Linux Public Login User Service (LXPLUS, https://lxplusdoc.web.cern.ch) cluster provided by CERN. User access to raw data is restricted to read-only. Raw data are additionally replicated to the so-called experiment space for backup. For the database instance, the CERN database service known as database on demand (DBoD) [16] is used, which in addition creates automatically a daily backup of the database. Authentication to the database for the ingestion mechanism is using a standard MySQL login path. Easy replication of the data outside campaigns is done via EOS's underlying xrootd (https://xrootd.slac.stanford.edu) storage server framework at CERN. In addition, chunked tar files for the whole campaign are created and copied to CASTOR[7] (http://castor.web.cern.ch), which automatically stores the files on tape. Control of the copy processes, tar file creation of the campaign data, and the ingestion mechanism are handled with virtual machines (VMs), created via CERN's OpenStack (https://www.openstack.org) cloud [17], [18]. On all VMs, EOS is available via a FUSE mount. These potentially long-running processes require Kerberos (http://www.kerberos.org) authentication via a generated key table.[8] This allows commands to be run for extended, possibly indefinite, periods of time in a tmux session or GNU screen.

2) Processing at Lisbon: The data transfer to the Lisbon servers happens in two stages. The first step involves data transfer to the gateway, where data are locally and temporarily stored. This synchronization is again done via rsync connecting to the rsync daemon on the main server. The second step is the transfer to Lisbon, also done via rsync connecting to the rsync daemon running on the Lisbon server. The Lisbon database is the main distribution point for monitoring the CLOUD experimental data from outside the CLOUD network. This is because access is easier for users since there is no need for port forwarding or individual user logins. It is therefore a simple single-step process. At Lisbon, a local database server is used to ingest the CLOUD data and provide global access to the databases. The rsync daemon and apache web service for data distribution are also running and made globally accessible through the firewall.

3) Storage Space and CLOUD Campaigns: Fig. 7 shows the data stored per campaign.[9] There are several types of CLOUD campaigns, namely, normal, technical, and purely cloudy

[7]Also provided by the CERN infrastructure for archiving physics data.

[8]Otherwise, the running process loses user authentication after the normally obtained Kerberos ticket expires.

[9]The actual (or logical) storage space is displayed, which refers to the storage space needed by the file and not the storage space allocated on a specific file system (i.e., the resident size).

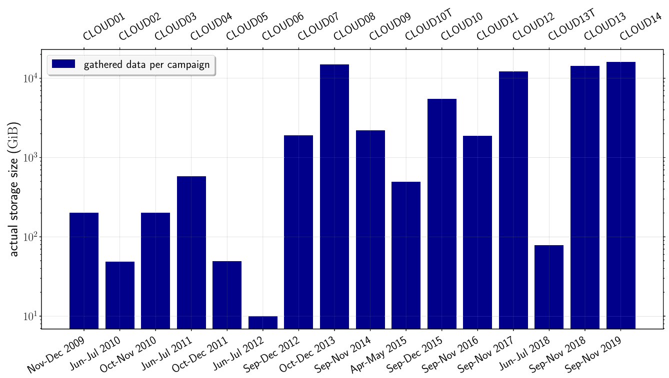

Fig. 7. Data stored for each CLOUD campaign. The increase in storage space is mainly due to the increasing number of mass spectrometers. For CLOUD8, however, the increase was due to CCD cameras.

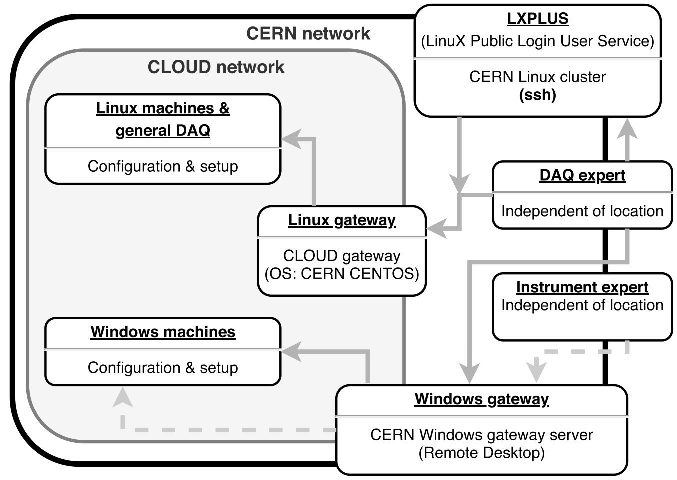

Fig. 8. Schematic showing the paths for remote access to the instrument computers. Access ways by the DAQ team are indicated by gray lines, while the dashed lines indicate those by instrument experts.

campaigns. Normal campaigns typically involve experiments on nucleation and growth of aerosol particles from precursor gases up to sizes where they can act as cloud condensation nuclei (CCN) (see [1], [19]). Cloudy campaigns include activation of CCN to cloud droplets or ice particles inside the CLOUD chamber (see [20], [21]) by means of adiabatic pressure reductions. Technical campaigns (denoted with a "T" suffix) are aimed at improving the understanding and performance of the CLOUD chamber and its analyzing instruments. The large increase in storage space over the more recent campaigns is due to an increasing number of mass spectrometers.

Taking for example the measurement period in 2018, CLOUD had an approximate data production of 8.9 TB during roughly 90 days of campaign. The incoming data to the central storage server averaged to 1.1 MB/s.

D. Remote Access and Instrument Control

Each CLOUD instrument requires expert intervention on a daily basis during experimental campaigns to monitor the data, adjust the operating parameters, and fix any problems that may arise. While much of this can be done by the instrument expert physically next to the instrument and its DAQ computer in the experimental zone, there are occasions where this is not possible, such as when the pion beam[10] is operating or when the expert is not on site at CERN. We have, therefore, set up a framework for remote access to the instrument computers, schematically shown in Fig. 8.

Since most of the instruments run on Windows, their remote access is handled via CERN Windows gateway servers, which requires a CERN computing account.[11] Access of the Windows gateway servers to the CLOUD network is configured via exceptions to the firewall rules. These firewall exceptions apply only to users with specific permissions set by the DAQ team.

Remote access to the Linux machines inside the CLOUD network is achieved as follows. If not inside the CERN network, an SSH [22] connection to CERN's LXPLUS is needed to enter the CERN network. From there, a connection to CLOUD's Linux gateways can be established, which provides access to the CLOUD network and the individual Linux machines. In some cases, a GUI is required, e.g., for the database setup and control applications; this is handled using X11 (https://www.x.org/wiki) forwarding[12] or the xrdp protocol (http://xrdp.org).

Similar to the other CLOUD instruments, remote access to the gas system and the so-called slow control system for the CLOUD facility are required. Those operate and read out the various systems, such as the thermometer strings, mixing fans, high voltage, and UV lamps. The gas and slow control systems are handled on CERN's TN.

E. Slow Control

The CLOUD slow control (CSC) system controls and monitors the facility for execution of the experiments. The system was redesigned in 2018 with three main design principles as follows:

1) implementation of a standard to control and monitor the CLOUD facility from the control room or remotely;
2) use of industrial components to provide better long-term maintenance.
3) improvement of the hardware protections to avoid equipment damage and minimize downtimes.

The CSC is a three-layer system. The bottom layer is the so-called "field," i.e., the hardware devices that need to be operated (e.g., motors, power supplies, and lamps) and instrumentation (e.g., temperature, pressure, humidity sensors) that monitors the state of the chamber as well as critical instrument parameters. The field communicates with the control layer by exchanging either raw analog/digital signals or through higher level communication protocols, such as TCP/IP over Ethernet. The middle layer is the control layer, which is in charge of receiving all incoming data and reacting according to a well-defined control process. In this layer, devices such as

[10]The π^+ beam is originating from protons provided by CERN's Proton Synchrotron (PS).

[11]Using CERN service accounts with limited access rights enables instrument experts without a valid CERN account to also remotely access the instrument computers.

[12]X11 forwarding has the advantage that no locally installed applications are needed. Where applications are installed locally, port forwarding is an alternative option.

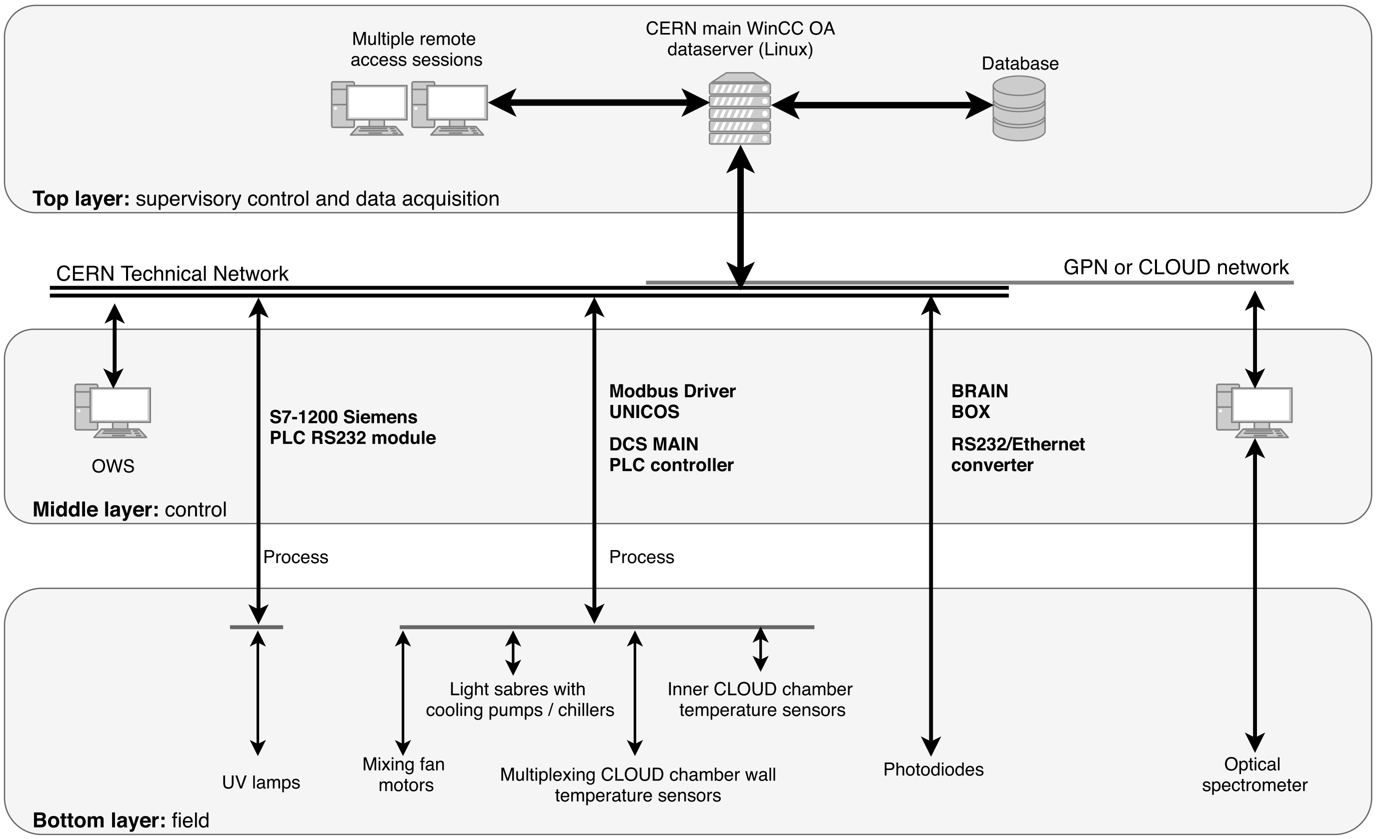

Fig. 9. Overview of the CSC system. At the bottom, the field layer is shown with the different hardware devices that are being readout and controlled. Above, it is the process layer, ensuring proper functioning of the devices. On the top, the SCADA system allows users to monitor and interact with the devices. To the right, an example subsystem is shown, which has not yet been implemented.

programmable logic controllers (PLCs) are deployed and interlocks are implemented, in order to prevent any operations that may damage the hardware. The top layer is the supervisory control and data acquisition (SCADA) system [23], [24] that provides the software to interact with the lower layers, through GUIs, as well as providing data visualization and archiving.

Fig. 9 shows an overview of the CSC system. The control interface involves: analog/digital signals that are fed into a PLC through its I/O modules. The photodiodes and UV lamps are controlled through an RS232 interface by a software module embedded into the control system and by a PLC, respectively. The optical spectrometer, which measures the spectral intensity of the illumination inside the CLOUD chamber, is at present a standalone system; the local data acquisition software is based on python-seabreeze (https://github.com/ap–/python-seabreeze) and communication with the control system relies on distributed information management (DIM) (http://dim.web.cern.ch) system [25], [26].

The software chosen for the top layer is a commercial SIEMENS SCADA toolkit—extensively used at CERN—called Simatic WinCC Open Architecture (WinCC OA; https://www.winccoa.com). WinCC OA is based on a distributed product, where quasi-independent processes, called managers, execute different tasks. Those managers do not need to run on the same machine and may be distributed on several computers running Windows or Linux as operating systems. This feature is useful since it does not restrict the system to a specific environment, making it more flexible and easier to implement.

Two CERN frameworks are in use above WinCC. The Joint COntrol Project (JCOP) [27], [28] and the UNified Industrial COntrol System (UNICOS) frameworks [29], [30] provide a full set of components allowing fast development of any detector control system. Tools, such as automatic PLC project generation, predefined widgets and faceplates, external database communication for archiving, specialized trend tools, or configurable alarm settings, are just a few examples of those features. A critical component in the CSC is the access control component. With the access control enabled, every user logs in with their personal account to perform any CSC action. Three authorization levels are in use: monitor, operator, and expert. Depending on the user's rights, different actions can be blocked or hidden to protect the detector integrity and guide the user.

The CSC works within a protected Ethernet network, the CERN's TN. Since some of the CSC data need to be rapidly available for the DAQ, these data are published, each 1–10 s (depending on the instrument) on the CERN network, and are subscribed to by the DAQ.[13] The communication protocol used for this is the DIP [8], [9]. The full set of CSC data is archived into an oracle database and can be retrieved through an interactive web application, customized specifically for CLOUD, based on pytimber (https://github.com/rdemaria/pytimber).

[13]From a networking point of view, firewall exceptions for the machine subscribing to the publication are set in place.

IV. Experimental Run in 2019 (CLOUD14) and Future Outlook

As mentioned earlier, during a typical measurement campaign, about 50 instruments are assembled and integrated into the CLOUD DAQ within a single setup week. The experimental run from September to end of November 2019 used the instruments listed in the Appendix.

The general CLOUD DAQ software implementation was tested to cope with data update blocks of a few kilobytes and update rates higher than 1 Hz. The main bottlenecks observed during the run were related to the instruments themselves, which had limited performance DAQ systems associated with the local hardware or software. In general, the central data ingestion tasks, although implemented in fast and optimized C++/Qt code, are resource demanding since they must handle data file formats that are not optimized. The data ingestion management system keeps 16 processes running in parallel and was able to cope with the full instrument set for the entire run period without difficulties. Time synchronization with the CLOUD reference time UTC was also a challenge since the automatic NTP update was not easy to set up in the diverse (and sometimes outdated) instrument computers, which additionally often lacked administrator or root access.

Since remote users were plotting and browsing data immediately after its acquisition, both the Lisbon and CERN "offline" systems were expected to be updated rapidly. The Lisbon replica could be maintained usually with less than 5-min delay with respect to the online system but suffered with occasional—but sometimes long duration— problems in the wide-area network from CERN to Lisbon. The replication of the CERN "offline" database also had an approximate delay of 5 min mainly due to a nonoptimized data transfer to the FUSE-mounted network filesystem EOS on the CLOUD gateways. Another issue related to the "offline" database at CERN was that access is considered cumbersome by the users because of the need for SSH tunneling and the nonstandard database port used at CERN.[14] A possible solution to this is related to an update of the internally developed CLOUD applications and using a standard MySQL login path combined with a dedicated, generic CERN service account exclusively used for data monitoring. This would allow the user to simply select the desired database, while authentication and connection are handled automatically in the background.

A major upgrade of the CLOUD experimental area currently ongoing at CERN includes a redesigned network infrastructure. This includes an optimized fiber optics network with improved routing, i.e., less intermediate routers and switches. It provides faster external monitoring by using a third source of replication at CERN that is completely detached from CERN standard solutions. This should be in physically close proximity to the experimental area in order to fully leverage the new fiber optics network. It also eliminates the previous weakness of a single point of failure for the external monitoring, namely, a single copper uplink that connects the entire CLOUD network to the CERN network. After the upgrade, faster setup times can also be expected because of the reduced amount of cabling. A redesigned electrical power grid separates individual instrument circuits and will guard against sector failures due to excessive power consumption of several instruments on the same circuit.

Frequent updates in the CLOUD DAQ instrument configuration also required supervision and manual intervention in the external database setup. This was needed mostly at the beginning of the run period when different instruments were being commissioned into the system or when a major change of run conditions[15] was made. This, however, is easily fixed by an update of the CLOUD applications with an optional automatic transfer of changes to the external database.

[14]Both of these measures are due to security restrictions.

V. Conclusion

We have described a novel DAQ system for the CLOUD experiment at CERN, which combines the data from a diverse set of autonomous instruments into a single integrated experiment. The DAQ system can be rapidly established over a short period of around five days at the start of each measurement campaign when the instruments are brought to the CLOUD chamber. The data from the autonomous instruments are combined through the UTC time stamp of their local computers. Although these show large drifts from an NTP time server, they are continually monitored and can be synchronized offline to around 10-ms precision. The DAQ framework is based on a database schema that avoids SQL-based database mirroring and the need for nonautomatic transactions, allowing the full system to run efficiently by exploiting standard MySQL solutions. Data are backed up and stored at multiple locations at CERN and Lisbon. Since CLOUD is a highly dynamic experiment—involving frequent adjustment of the experimental conditions—it is vital that the DAQ system provides rapid access to and analysis of the experimental data. This is provided by custom high-quality plotting routines that provide online monitoring at a 30-s refresh interval inside the CLOUD network. The plotting software is highly flexible, allowing users to customize the displayed data according to their specific needs and perform rapid online analyses of the experiment in progress, helping to guide when experimental transitions can be made. A parallel capability is provided for similar monitoring and analysis by remote researchers—with a 5-min delay—using a mirrored external site at Lisbon or CERN. The DAQ system has proved to be a flexible and robust framework that successfully meets the special requirements of the CERN CLOUD experiment.

Appendix
CLOUD14 Instrument List

Instruments used during the CLOUD measurement campaign from September to November 2019. The naming is as they are known by the aerosol community.

[15]Such a major change usually happens three or four times during a measurement campaign. This basically implies a switch in the studied chemical system, e.g., moving from a simulated marine environment to an anthropogenic environment.

1) *INP and CCN Counters:*[16] PINCii (Portable Ice Nucleation Chamber), mINKA (Mobile Ice Nucleation Instrument of the KArlsruhe Institute of Technology), PINE (Portable Ice Nucleation Experiment) with DPM (Dew Point Mirror), SPIN (SPectrometer for Ice Nuclei), CCNc (Cloud Condensation Nuclei Counter), nano-HTDMA (Hygroscopic Tandem Differential Mobility Analyser), SIMONE (scattering intensity measurement for the optical detection of ice, translated from the German name Streulicht-IntensitätsMessungen zum optischen Nachweis von Eispartikeln), and WELAS (WhitE-Light Aerosol Spectrometer).
2) *Mass Spectrometers and Related:* ±API-ToF (Atmospheric Pressure Interface Time Of Flight mass spectrometer), CI (Chemical Ionization)-Orbitrap, (NH_3^--)CI-API-ToF with TD-DMA (Thermal Desorption Differential Mobility Analyzer), EESI (Extractive ElectroSpray Ionization)-API-ToF, PTR3 (Proton Transfer Reaction time of flight mass spectrometer), PTRS, FIGAERO (Filter Inlet for Gases and AEROsols) (Br^-/I^--)CI-ToF, and AMS (Aerosol mass spectrometer).
3) *Particle Instruments:* SMPS (Scanning Mobility Particle Sizer), nano-SMPS, long-SMPS, CPC (Condensation Particle Counter), CPC 2.5, CPC 3010, PSM (Particle Size Magnifier), NAIS (Neutral cluster and Air Ion Spectrometer), CIC (Cluster Ion Counter), nano-SEMS (Scanning Electrical Mobility Spectrometer), and LDMA (Long-column Differential Mobility Analyzer).
4) *Gas Measurement:* O_3 monitor, SO_2 monitor, NO_2 monitor, NO monitor, CO monitor, NH_3 monitor, DPM, and In situ-TDL (Tuneable Diode Laser)-Hygrometer with absolute pressure sensor.
5) *Temperature Related:* Horizontal TC (ThermoCouple) string, horizontal PT100 (Platinum resistance Thermometer) calibration string for TC string, PT100 string, thermal housing PT100 sensors, chamber temperature control unit, and additional temperature control unit for instruments.
6) *Light Related:* Light spectrometer, 5 × Photodiodes, excimer laser, with cooling unit, Light Sabers: 1, 3, and 4 with cooling units, and fiber-optic UV system.
7) *Others:* Gas & Expansion control system (over 800 parameters), 2 × Mixing fans, 2 × High voltage power supply for Electric field generation & Grounding Box, GCR (Galactic Cosmic Ray) counter, Hodoscope (array of 18 × phototubes), Beam counter (2 × scintillators), π^+ beam (collimator / stopper setting & related parameters), 4 × Raspberry Pi cameras, CHARGE (CHarged Aerosol GEnerator), Evaporative H_2SO_4 aerosol generator, and filter sampling.

[16]INP (Ice Nucleating Particle) & CCN counters.

References

[1] J. Kirkby *et al.*, "Role of sulphuric acid, ammonia and galactic cosmic rays in atmospheric aerosol nucleation," *Nature*, vol. 476, pp. 429–433, Aug. 2011.
[2] U. Epting *et al.*, "Computing and network infrastructure for controls CNIC," in *Proc. 10th Int. Conf. Accel. Large Exp. Phys. Control Syst. (ICALEPCS)*, 2005. [Online]. Available: https://cds.cern.ch/record/806508
[3] J. T. Moscicki and M. Lamanna, "Prototyping a file sharing and synchronization service with Owncloud," *J. Phys. Conf. Ser.*, vol. 513, no. 4, 2014, Art. no. 042034.
[4] L. Mascetti, H. G. Labrador, M. Lamanna, J. Mościcki, and A. Peters, "CERNBox + EOS: End-user storage for science," *J. Phys., Conf. Ser.*, vol. 664, no. 6, Dec. 2015, Art. no. 062037.
[5] D. Vadala, *Managing RAID on Linux*. Sebastopol, Calif: O'Reilly Media, 2002.
[6] A. Tridgell and P. Mackerras, "The Rsync algorithm," Dept. Comput. Sci., Comput. Sci. Lab., Austral. Nat. Univ., Canberra, ACT, Australia, Tech. Rep. TR-CS-96-05, 1998. [Online]. Available: https://rsync.samba.org/tech_report/
[7] H. Junninen *et al.*, "A high-resolution mass spectrometer to measure atmospheric ion composition," *Atmos. Meas. Techn.*, vol. 3, no. 4, pp. 1039–1053, Aug. 2010.
[8] W. Salter *et al.*, "Dip description," LHC Data Interchange Work. Group (LDIWG), CERN, Geneva, Switzerland, Tech. Rep., 2004. [Online]. Available: https://cern.ch/dip/
[9] B. Copy, E. Mandilara, I. P. Barreiro, and F. Varela, "Monitoring of CERN's data interchange protocol (DIP) system," in *Proc. Int. Conf. Accel. Large Exp. Control Syst. (ICALEPCS)*, Barcelona, Spain: JACoW, Jan. 2018, pp. 1797–1800, paper THPHA162. [Online]. Available: http://jacow.org/icalepcs2017/papers/thpha162.pdf
[10] D. L. Mills, "Internet time synchronization: The network time protocol," *IEEE Trans. Commun.*, vol. 39, no. 10, pp. 1482–1493, Oct. 1991.
[11] D. Mills, J. Burbank, and W. Kasch, "Network time protocol version 4: Protocol and algorithms specification," Internet Eng. Task Force (IETF), Tech. Rep. rfc5905, Jun. 2010.
[12] D. L. Mills, *Computer Network Time Synchronization*, D. L. Mills, Ed. Boca Raton, FL, USA: CRC Press, Dec. 2017.
[13] D. Mills, "Simple network time protocol (SNTP) version 4 for IPv4, IPv6 and OSI," Internet Soc., Netw. Working Group, Tech. Rep. rfc4330, Jan. 2006.
[14] A. J. Peters and L. Janyst, "Exabyte scale storage at CERN," *J. Phys., Conf. Ser.*, vol. 331, no. 5, Dec. 2011, Art. no. 052015.
[15] A. Peters, E. Sindrilaru, and G. Adde, "EOS as the present and future solution for data storage at CERN," *J. Phys., Conf. Ser.*, vol. 664, no. 4, Dec. 2015, Art. no. 042042.
[16] R. G. Aparicio and I. C. Coz, "Database on demand: Insight how to build your own DBaaS," *J. Phys., Conf. Ser.*, vol. 664, no. 4, Dec. 2015, Art. no. 042021.
[17] T. Bell, B. Bompastor, S. Bukowiec, and M. F. Lobo, "Scaling the CERN OpenStack cloud," *J. Phys., Conf. Ser.*, vol. 664, no. 2, Dec. 2015, Art. no. 022003.
[18] M. Denis, J. C. Leon, E. Ormancey, and P. Tedesco, "Identity federation in OpenStack—An introduction to hybrid clouds," *J. Phys., Conf. Ser.*, vol. 664, no. 2, Dec. 2015, Art. no. 022015.
[19] D. Stolzenburg *et al.*, "Rapid growth of organic aerosol nanoparticles over a wide tropospheric temperature range," *Proc. Nat. Acad. Sci. USA*, vol. 115, no. 37, pp. 9122–9127, Aug. 2018.
[20] E. Järvinen *et al.*, "Observation of viscosity transition in α-pinene secondary organic aerosol," *Atmos. Chem. Phys.*, vol. 16, no. 7, pp. 4423–4438, Apr. 2016.
[21] K. Ignatius *et al.*, "Heterogeneous ice nucleation of viscous secondary organic aerosol produced from ozonolysis of α-pinene," *Atmos. Chem. Phys.*, vol. 16, no. 10, pp. 6495–6509, May 2016.
[22] T. Ylonen, "The secure shell (SSH) protocol architecture," Internet Soc., Netw. Working Group, Tech. Rep. rfc4251, Jan. 2006.
[23] A. Daneels and W. Salter, "What is SCADA?" in *Proc. 7th Int. Conf. Accel. Large Exp. Phys. Control Syst.*, vol. 991004, 1999, pp. 339–343, 1999. [Online]. Available: https://inspirehep.net/conferences/971887
[24] A. Daneels and W. Salter, "Selection and evaluation of commercial SCADA systems for the controls of the CERN LHC experiments," in *Proc. 7th Int. Conf. Accel. Large Exp. Phys. Control Syst.*, vol. 991004, 1999, pp. 353–355. [Online]. Available: https://inspirehep.net/conferences/971887
[25] C. Gaspar and M. Dönszelmann, "DIM—A distributed information management system for the DELPHI experiment at CERN," CERN, Geneva, Switzerland, Tech. Rep. DELPHI-94-4 DAS 148, 1994. [Online]. Available: https://inspirehep.net/literature/1661387
[26] C. Gaspar, M. Dönszelmann, and P. Charpentier, "DIM, a portable, light weight package for information publishing, data transfer and interprocess communication," *Comput. Phys. Commun.*, vol. 140, nos. 1–2, pp. 102–109, 2001.

[27] D. Myers, “The LHC experiments’ joint controls project, JCOP,” in *Proc. 7th Int. Conf. Accel. Large Exp. Phys. Control Syst.*, vol. 991004, pp. 633–635, 1999. [Online]. Available: https://inspirehep.net/conferences/971887

[28] O. Holme, M. Gonzalez-Berges, P. Golonka, and S. Schmeling, “The JCOP framework,” CERN, Geneva, Tech. Rep. CERN-OPEN-2005-027, Sep. 2005. [Online]. Available: https://cds.cern.ch/record/907906

[29] F. Bernard, E. Blanco, A. Egorov, P. Gayet, H. Milcent, and C. H. Sicard, “Deploying the UNICOS industrial controls framework in multiple projects and architectures,” in *Proc. 10th Int. Conf. Accel. Large Exp. Phys. Control Syst. (ICALEPCS)*, 2005. [Online]. Available: https://accelconf.web.cern.ch/ica05/proceedings/html/contents.htm

[30] P. Gayet and R. Barillère, “UNICOS a framework to build industry-like control systems principles methodology,” in *Proc. 10th Int. Conf. Accel. Large Exp. Phys. Control Syst. (ICALEPCS)*, 2005. [Online]. Available: https://accelconf.web.cern.ch/ica05/proceedings/html/contents.htm

Stefan K. Weber was born in 1988. After an early start in industry, he decided to move to university to study. He received the B.Sc. degree in physics and the M.Sc. degree in physics from the Technical University of Munich, Munich, Germany, in 2014 and 2017, respectively. He is currently pursuing the Ph.D. degree with CERN, Geneva, Switzerland.

Giovanna Lehmann Miotto was born in 1972. She received the M.Sc. degree in physics from the University of Milan, Milan, Italy, in 1996, and the Ph.D. degree in physics from the University of Bern, Bern, Switzerland, in 2000.

She is currently a Staff with CERN, Geneva, Switzerland.

João Almeida received the degree in geophysics from the University of Lisbon, Lisbon, Portugal, in 2007.

He worked as a Data Scientist with the Portuguese Meterological Institute, Lisbon, and the European Centre for Medium-Range Weather Forecasts, Reading, U.K. He has been involved in the CLOUD experiment at CERN, Geneva, Switzerland, as a Consultant, since 2015. He is currently a Freelancer with activities focus on GPU programming and blockchain technologies.

Pascal Herve Blanc was born in August 1958.

He was an Electromechanical Technician with Pechiney Electrometallurgie, Paris, France. Since December 2013, he has been an Electromechanical Technician (EP/DT/DI) with CERN, Geneva, Switzerland.

António Dias was born in Portugal in 1989. He received the master’s degree in physics engineering from the Instituto Superior Técnico, Lisbon, Portugal, in 2011, and the Ph.D. degree from the University of Lisbon, Lisbon, in 2020.

He was a CLOUD-TRAIN EU Marie Curie Fellow from 2013 to 2015.

Giulio Malaguti was born in 1992. He received the degree Higher Technician Diploma for Automation and Mechatronic Systems from the Higher Technical Institute (ITS) for Automation and Packaging, Bologna, Italy, in 2015.

He is currently a Staff with CERN, Geneva, Switzerland.

Hanna E. Manninen received the M.Sc. degree in aerosol measurement and instrumentation and the Ph.D. degree in air ion measurement techniques from the Department of Physics, Faculty of Science, University of Helsinki, Helsinki, Finland, in 2007 and 2011, respectively.

She is a Ph.D.-qualified experimental physicist with 13 years’ experience in environmental observations. She has solid principles and practice of environmental science and technology, including instrument and protocol development.

Joschka Pfeifer was born in 1991. He received the M.Sc. degree in meteorology from Johann Wolfgang von Goethe University, Frankfurt, Germany, in 2018, where he is currently pursuing the Ph.D. degree in geosciences.

Sylvain Ravat received the master’s degree in mechanical and automated production engineering from ENIT, Tarbes, France, in 1998, and the master’s degree in industrial automation from INSA, Lyon, France, in 1999.

Since 2004, he has been a Control Software Engineer with the Detector Technologies (DT) Group, Experimental Physics (EP) Department, CERN, Geneva, Switzerland. He has participated in the development of the control and safety systems for LHC Experiments (Alice, ATLAS, CMS, and LHCb) magnets.

Antti Onnela received the M.Sc. degree in mechanical engineering from the Helsinki University of Technology, Espoo, Finland, in 1993.

After an early career in manufacturing industries in Finland and Germany, in 1992, he started at CERN, Geneva, Switzerland, on the design of particle tracking detectors for the CMS experiment. Alongside with his continuing work on particle tracker mechanics and cooling, he is the Project Engineer and a Safety Officer of the CLOUD experiment. He is currently a Section Leader with the Detector Technologies Group, CERN. He chairs the Computer-Aided Engineering Committee that selects CAE software tools and sets policies for their use at CERN.

Serge Mathot received the Ph.D. degree in applied physics from the University of Namur, Namur, Belgium, in 1992.

He has been a Physicist with CERN, Geneva, Switzerland, since 1995. His main activities include the Technical Coordinator for the CLOUD experiment, the CERN Project Leader for the CERN/INFN join project MACHINA (Movable Accelerator for Cultural Heritage In situ Non destructive Analysis), and the Project Leader for Experimental LInac for Surface Analysis (ELISA), a 2-MeV proton accelerator for exhibition at the CERN Science Gateway.

Jasper Kirkby was born in Prestwich, U.K., in 1948. He received the M.A. degree in physics from the Oriel College, Oxford University, Oxford, U.K., in 1969, and the Ph.D. degree in high energy physics from the Westfield College, University of London, London, U.K., in 1972.

He has carried out experimental research at the Stanford Linear Accelerator Center (SLAC) from 1972 to 1984. He has been with the European Centre for Nuclear Physics (CERN), Geneva, Switzerland, since 1984.

Dr. Kirkby received the Benjamin Y.H. Liu Award of The American Association for Aerosol Research (AAAR) in 2019 for originating and leading the CLOUD experiment at CERN.

António Tomé was born in Monte Estoril-Cascais, Portugal, in 1963. He received the Licentiate degree in geophysical sciences with the University of Lisbon, Lisbon, Portugal, in 1989, and the Ph.D. degree in physics from the Universidade da Beira Interior, Covilhã, Portugal, in 1997.

He was a Researcher with the Instituto Geofísico Infante D. Luís, Lisbon, under supervision of Prof. José Pinto Peixoto from January 1989 to February 1990. Since 1990, he has been with the Universidade da Beira Interior.

António Amorim received the Ph.D. degree in nuclear and particle physics from the University of Lisbon, Lisbon, Portugal, in 1991.

He is currently a Professor with the Physics Department, University of Lisbon. He was involved in the development DAQ and information systems at CERN, Geneva, Switzerland, first for the ATLAS and later for the CLOUD collaborations. His main research is on instrumentation for astrophysics, associated with infrared observation instruments for ESO.