Provenance–aware workflow for data quality management and improvement for large continuous scientific data streams

Jitendra Kumar*, Michael C. Crow*, Ranjeet Devarakonda*, Michael Giansiracusa*, Kavya Guntupally*, Joseph V. Olatt†, Zach Price†, Harold A. Shanafield III*, Alka Singh‡

*Environmental Sciences Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA
Email: jkumar@climatemodeling.org, crowmc@ornl.gov, devarakondar@ornl.gov, giansiracumt@ornl.gov, guntupallyk@ornl.gov, shanafieldha@ornl.gov
†Information Technology Services Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA
Email: olattjv@ornl.gov, pricezt@ornl.gov
‡Electrical and Electronics Systems Research Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA
Email: singhar@ornl.gov

***Abstract*—Data quality assessment, management and improvement is an integral part of any big data intensive scientific research to ensure accurate, reliable, and reproducible scientific discoveries. The task of maintaining the quality of data, however, is non-trivial and poses a challenge for a program like the Department of Energy's Atmospheric Radiation Measurement (ARM) that collects data from hundreds of instruments across the world, and distributes thousands of streaming data products that are continuously produced in near-real-time for an archive 1.7 Petabyte in size and growing. In this paper, we present a computational data processing workflow to address the data quality issues via an easy and intuitive web-based portal that allows reporting of any quality issues for any site, facility or instruments at a granularity down to individual variables in the data files. This portal allows instrument specialists and scientists to provide corrective actions in the form of symbolic equations. A parallel processing framework applies the data improvement to a large volume of data in an efficient, parallel environment, while optimizing data transfer and file I/O operations; corrected files are then systematically versioned and archived. A provenance tracking module tracks and records any change made to the data during its entire life cycle which are communicated transparently to the scientific users. Developed in Python using open source technologies, this software architecture enables fast and efficient management and improvement of data in an operational data center environment.**

***Index Terms*—scientific data workflows, data quality, provenance, atmospheric science,**

I. Introduction

Rapid growth in observational technologies, data management, and sharing infrastructure has enabled a new era of big data enabled scientific discovery in a wide range of disciplines. However, ensuring the high quality of the data is crucial for accurate and reliable scientific research and decision making. Management of data quality presents specific challenges for environmental monitoring networks that operate in large numbers of distributed facilities in remote regions of the world, like the U.S. Department of Energy's Atmospheric Radiation Measurement (ARM) program (https://arm.gov). Ensuring high quality of data to scientific community requires not only the assessment and documentation of data quality issues but also appropriate data reprocessing to address and improve the data quality. It is equally important is to capture the provenance of data at every step of its life cycle that are comprehensive, timely, and transparently communicated to the data users. Performing data processing at the scale of a scientific data center like ARM poses a big data challenge. Diversity of sensors and instruments also call for an comprehensive metadata and data processing automation. The sheer volume of the data necessitates state-of-the-art parallel processing, efficient data movement, I/O, and provenance tracking and management. In this paper we describe a provenance-aware workflow for data quality improvement of continuously growing atmospheric science data streams within ARM's Petabyte scale archive.

978-1-7281-0858-2/19/$31.00 ©2019 IEEE

II. Atmospheric Radiation Measurement

A. Introduction to ARM program

For the 30 years and counting since its inception in 1990 [1], the ARM program has been collecting observations across the globe to advance the robust predictive understanding of Earth's climate and environmental systems and to inform the development of sustainable solutions to the Nation's energy and environmental challenges. Deployed across three atmospheric observatories, three mobile facilities and various aerial facilities, data are continuously being collected from 480 scientific instruments and sensors. With a 1.7 Petabyte and growing archive of data, ARM makes available to the scientific community, publicly and freely, approximately 8218 data streams of which 1220 are actively growing in near-real-

978-1-7281-0858-2/19/$31.00 ©2019 IEEE

time with data streaming from the instruments around the world.

B. Assessment and Management of Data Quality

These instruments for atmospheric observations, however, are prone to data quality issues due to the challenging operating conditions in the field , sensor failures, the need for recalibration, and etc. ARM's Data Quality Office, established in year 2000, coordinates detection and reporting of data quality for all data streams [2]–[4].

Data Quality Reports (DQR) can be submitted by data quality analysts, instrument mentors, operations personnel, or data users and are saved within a consistent and searchable PostgreSQL database. However, with large numbers of instruments and streaming data stream maintained by the program, DQRs can be frequent and numerous. Figure 1 shows the reported data quality issues for 20 ARM instruments over time. Some instruments are more prone to data quality issues than others, often due to the nature of deployed sensors, and it is essential that the data be reprocessed to ensure the availability of high quality continuous time series. Within the ARM program, reprocessing of data is conducted whenever either a correctable data quality issue is identified or an improved processing algorithm is available. However, the volume and diversity of data poses a challenge.

Historically the complexity of data reprocessing to address the quality issues, data size, and volume has limited the improvements made to correct for the data quality issue (Figure 2). For example, 30 minute resolution energy balance Bowen Ratio (30EBBR) instruments that have been in operation since 1993 have largest number of DQRs, most of which have not been addressed. Surface meteorology (MET) observations, which are one of the most highly used datasets, had large number of reported quality issues, only a few of which have been corrected. In contrast, radiation measurements (RAD) instruments had very few quality issues, most of which were corrected.

III. Objectives

The objective of our study was to develop an efficient computational workflow for processing atmospheric science data sets to address data quality issues. This workflow was designed to 1) allow for easy and intuitive data quality issue reporting system to encourage data correction in addition to reporting; 2) automate data reprocessing to reduce human intervention and thus maximize efficiency; 3) process a large volume of datasets in parallel; 4) provide systematic process verification and version control; 5) capture provenance information at each step; and 6) clearly communicate any updates to the data to all stakeholders.

IV. Workflow for data quality improvement

ARM data reprocessing to address data quality issues was designed as an end-to-end workflow from collection of data quality issues and concern, reprocessing for data quality improvement, review for accuracy, versioning and archival and user notification (Figure 3). The computational workflow for reprocessing data to address data quality issues was developed in Python programming language. The software framework was designed for computational performance and cross-platform compatibility.

A. Reporting Data Quality Issues

Data quality issues are identified through a number of mechanisms including data quality assessment tools, instrument mentors, project scientists and data users. A web-based DQR submission portal was designed to enable easy and comprehensive reporting of data quality issues for any ARM instrument or data stream (Figure 4). Portal consists of a complete database of all current and historical instruments and data streams across ARM facility allowing data quality report to be submitted for any site, facility, instrument or data stream at the granularity of specific affected variable and for the selected time period. In addition to the detailed description of the quality issue and affected variables, the portal also allows for corrective update and maintenance actions to be suggested by the instrument experts. Suggestions for correcting the data can be provided in form of symbolic equations or as descriptive text. The auto-complete feature of the portal ensures easy and correct variables names to be used in the symbolic equation, and also detects and captures dependencies among affected variables. Each DQR is assigned an unique identifier (DQRID) and is recorded within a PostgreSQL database.

B. Automating data processing

An end-to-end computational workflow was developed for data quality improvement to address the reported data quality issue.

1) Data staging and processing environment: Raw data from all ARM instruments are versioned and saved in deep archive on the High Performance Storage System (HPSS) [5] at Oak Ridge Leadership Computing Facility [6]. An automated process was developed to query the database for any data quality issue iand based on it's unique DQR ID, identify all raw data files impacted by the reported issues; the impacted datasets are then retrieved from the archive. Two different methods for data retrieval were implemented: 1) automated retrievals using HPSS Hierarchical Storage Interface (HSI) [5] which can be used across most of the computational systems within ARM Data Center, but, performs a sequential retrieval of files; and 2) a Globus Online [7], [8] based fast and parallel transfer mechanism suitable for large volumes of data.

2) Instrument data dictionaries: Raw data from each individual instrument, which number in hundreds across ARM facility, are all often different depending on sensors, data characteristics and manufacturers. The raw data format ranges from space or comma separated ASCII, binary, hex etc. Ability to automate the processing of datasets require a good machine readable metadata and descriptors for each instrument. Traditional processing workflows were custom designed for individual instruments and thus lead to rigid workflow that required custom developments to address any quality issues.

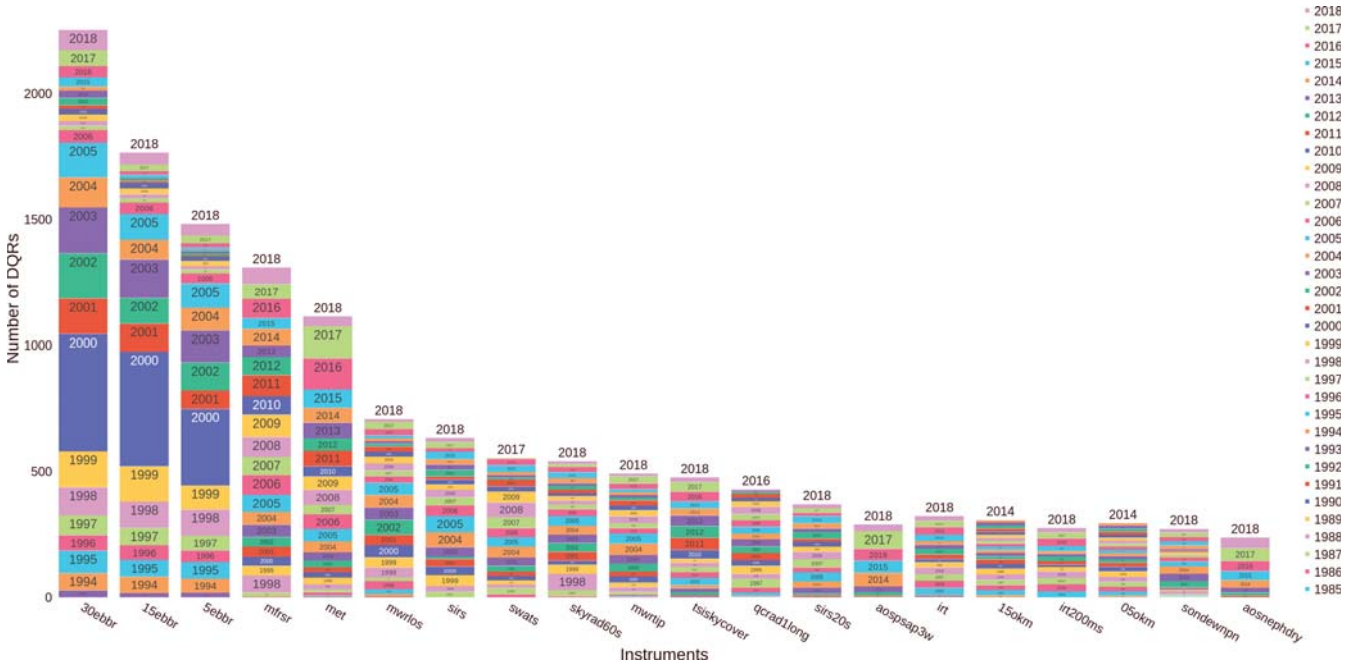


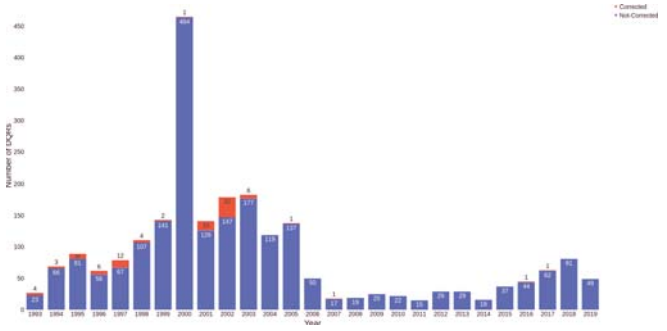
Fig. 1. DQRs for select 20 instruments over the period 1993-2019 show that some instruments are more prone to data quality issues than the others. Time series of DQRs show that data quality issues at many of the instruments are repetitive and recurring.

As part of automated workflow, detailed data dictionaries were designed and saved in machine readable JSON formats that allowed for automated processing of data. Data dictionary encodes the format of raw instrument data, numerical procedure to calculated derived variables, and ARM standard variable names and units (Figure 5). Data dictionaries also capture any historical change or update to instrument data format or observations. Figure 5 shows an example data dictionary for surface meteorology (MET) instruments the contains the column-wise mapping of data in raw ASCII format, expected data type, units and names for standard primary and derived variables. Library of comprehensive data dictionaries are the critical to enable automation of data quality control and improvements at scale across large observational facility like ARM.

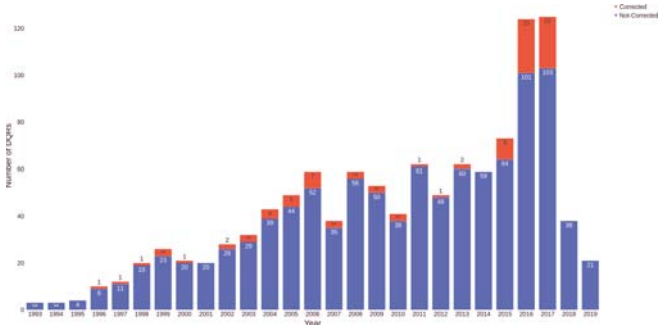
3) Symbolic equations processing: As part of automated data processing framework, a capability for symbolic equation processing was developed using Python `SymPy` [9]. Equations for calculating any primary or derived variables are derived from instrument data dictionaries or provided by instrument experts as part of the associated DQR. Equations provided with DQRs are given precedence over ones from data dictionaries. Framework was designed to ensure the correctness of associated variables to be processed for any given data stream. When solving system or set of equations, the framework was designed to detect and handle and variable dependence and/or conflict. If primary variable is changed as part of a data quality issue, all dependent derived variables were recalculated to ensure correctness of the entire data stream as part of the processing. Framework was implemented and tested for a range of scenarios experienced within ARM datasets for correctness and accuracy before being deployed operationally. Ability to process symbolic equations allowed for automation of data improvement workflow, avoid need for any custom development and processing by an analyst.

4) High performance computing for big data processing: For efficient processing of large volumes of data, capability for parallel data processing was developed. Most ARM time series data are packaged in one or more files per day allowing for their simultaneous processing in a embarrassingly parallel fashion. Two parallel processing framework were implemented within the Python based software framework, to allow for use and execution on small shared memory computing environment to medium to large scale distributed memory computing environments. For shared memory environment, parallelism was achieved using Python `multiprocessing` while Python `Dask` [10] was used for parallel data processing on distributed memory compute clusters. For task requiring processing for large volume of data, data transfers times are significant and system was designed to overlap data staging and processing to allow processing of data as soon as they become available.

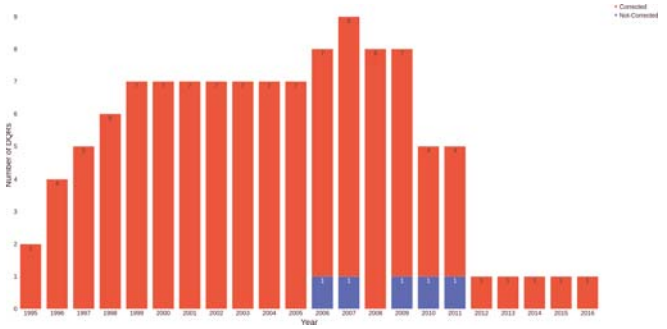
5) Data review: Review of the processing performed on the data is critical to ensure that the actions undertaken accurately and satisfactorily addressed the data quality issues reported in the associated DQR. A formal and comprehensive data review was conducted for each corrected and processed datasets, comparing old and new data products to identify variables affected, compute statistics for change and generate interactive plots to visualize the data. Data review was one of the only



(a) 30 minute Energy Balance Bowen Ratio (EBBR) instruments are one of the longest running core instruments within ARM and while it had a number of reported data quality issues every year, most of them have not been addressed.



(b) Surface Meteorology (MET) are another core instruments at ARM facilities that with frequent reported data quality issues, and many of them have been addressed in recent years as data quality improvement infrastructure has improved.



(c) Radiation Measurements (RAD) instruments have had fairly small number of data quality issues, most of which have been addressed.

Fig. 2. Only a small fraction of reported quality issues are corrected (shown in red) while the majority remains uncorrected (shown in blue). The fraction of corrected vs not-corrected are highly variable across different instruments and are often determined by availability of resolution for the issue, process complexity, and data volume.

step that was designed to require a human intervention, in form of review either by the instrument expert or the data analyst. While the workflow was designed for automation, the free-form description section of the DQR may at times contain special instructions or information that may have been missed by the automation workflow. Data review was designed to provide the oversight to ensure the accuracy of the data. If the data review check is successful, it triggers the execution of the rest of the workflow, however in case an issue is identified the

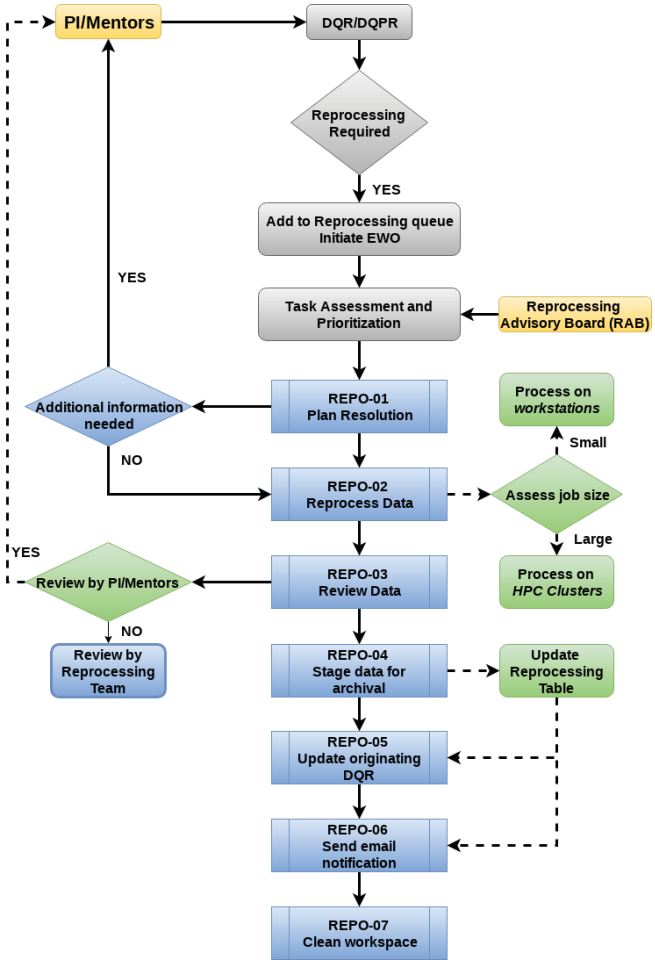


Fig. 3. End-to-end workflow to reprocess data for quality improvement

processing task is assigned to a data analyst.

V. Data Versioning and Archival

Once an updated version of data with improved data quality is prepared, it is formally assigned a version for tracking. However, versioning for complex data like those in ARM pose unique challenges which led to development of new data versioning system deployed across the facility [11]. Data reprocessing to data quality improvement, however, at times leads to versioning conflict scenarios. Versioning module of the workflow extends the current versioning scheme [11].

Standard ARM filename and versioning includes site, facility, instrument and start time of observation time series in filename. However, length of time series contained within the file are not reflected in the filename and thus creating conflict when data are reprocessed. ARM file standards also allow for any arbitrary number of files within a day during a real time processing, while reprocessing for data quality improvements on past data often create longer continuous time series and thus few files. To identify and address these conflicts all affected files (old and new) were checked for the start and end of the time series. In particular following scenarios occur frequently:

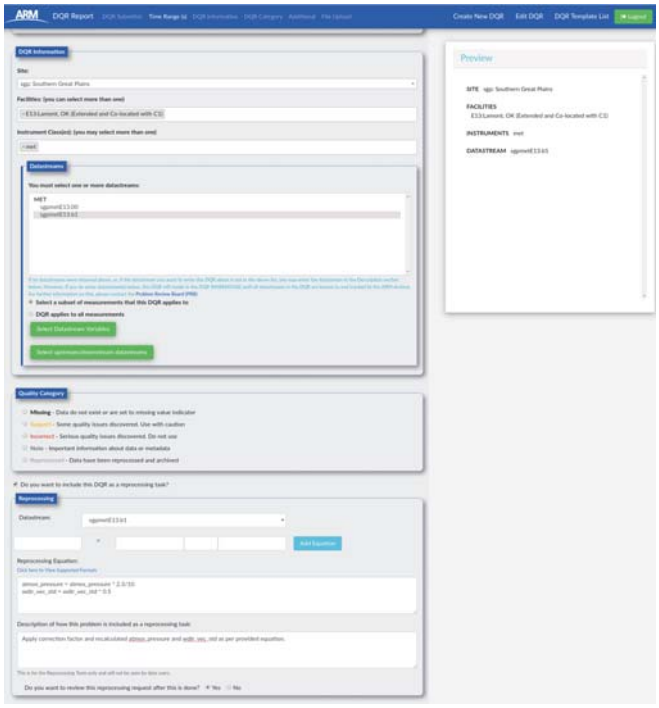


Fig. 4. DQR Submission Tool allows reporting of data quality issues for any observational site, facility and instrument to the granularity of individual variable within the data files. Suggestion for data correction can be provided in form of symbolic equations which be used for automated processing tools to apply the correction.

1) Near-real-time processing may create multiple partial day files within a day (ex. fileA.v0: 00:00:00–12:00:00 and fileB.v0: 12:00:01–24:00:00) if the observations from the remote instruments as observations arrive at the data center in chunks due to network bandwidth or some other issue. However, the reprocessing for the data may create a single file with continuous time series (ex. fileA: 00:00:00–24:00:00). While the filename of the newly created file would be same as one of the existing files and will be assigned the correct incremental version number (ex. fileA.v1: 00:00:00–24:00:00), the rest of the old files (ex. fileB: 12:00:01–24:00:00) from the day would identified and marked to be obsolete and marked unavailable in future.
2) File versioning tracks the versions of a filename, but during the process of reprocessing it's possible for new filenames to be created. For example, if data in original fileA.v0: 01:00:00–24:00:00 was reprocessed to correct the quality issue during first hour of the day that was excluded for the original processing, a new fileB.v0: 00:00:00–24:00:00 will be created and while no filename conflict occur with fileA.v0, fileA.v0 must be deleted and marked unavailable to avoid conflicting time series data.

Once the file has been assigned the correct name and version, it is archived in the deep archive on HPSS and a


```
{
  "header":{
    "num_rows":4,
    "row_1":"data format,,,",
    "row_2":"variables",
    "row_3":"units",
    "row_4":"type"
  },
  "data":{
    "timestamp":{
      "column":0,
      "format":"yyyy-mm-dd hh:mm:ss.d",
      "cdf_var":[
        "base_time",
        "time_offset",
        "time"
      ]
    },
    "RN":{ },
    "batt_volt":{ },
    "PTemp":{ },
    "Pressure_kPa":{
      "column":4,
      "type":"smp",
      "cdf_var":"atmos_pressure"
    },
    "Temp_C_Avg":{ },
    "Temp_C_Std":{ },
    "RH_Avg":{ },
    "RH_Std":{ },
    "Vap_Pressure_kPa_Avg":{ },
    "Vap_Pressure_kPa_Std":{ },
    "WS_MS_S_WVT":{ },
    "WS_MS_U_WVT":{ },
    "WindDir_DU_WVT":{ },
    "WindDir_SDU_WVT":{
      "column":14,
      "type":"wvc",
      "cdf_var":"wdir_vec_std"
    },
    "rain_mm_Tot":{ },
    "rain_mm_min_corrected":{ },
    "WS_Slope":{ },
    "WS_Offset":{
      "column":18,
      "type":"smp",
      "cdf_glob":"wind_speed_offset"
    },
    "TBRG_SN":{
      "column":19,
      "type":"smp",
      "cdf_glob":"serial_number"
    },
    "RainCoefA":{
      "column":20,
      "type":"smp",
      "cdf_glob":"tbrg_precip_corr_info",
      "input_for":[
        "tbrg_precip_total_corr"
      ]
    },
    "RainCoefB":{ }
  }
}
```


Fig. 5. Data dictionary for MET instrument contains information about the raw data format, standard variable names, units, variable dependencies and other metadata in a JSON format.

copy is placed on new data cache while the older data are made unavailable. ARM's comprehensive database of all file holdings are also updated as a part of the archival step.

VI. Data provenance tracking

Accuracy and reproducibility is essential component of sound and reliable scientific research and analysis. Capturing rigorous provenance information at every step of data life cycle is critical to allow reproducible science. Data reprocessing workflow for data quality improvement modifies the data from its original version and its paramount that the provenance of the data be comprehensively captured. While the technical details of the data quality issue and it is implication are detailed in the DQR, the reprocessing workflow was designed to log details of every step of the processing including versioned filenames of each data file processed, variables recalculated, symbolic equations applied for processing, softwares including versions used in processing, computing systems used etc. A subset of provenance information relevant for scientific users of the data was also appended to the DQR, that is distributed along with the data.

VII. Communicating data quality changes

Clear and timely communication of the data quality changes to all stakeholders including scientific users is important to maintain the research integrity. While the description of data quality issues and remedial actions taken to address them are available publicly as part of the DQR associated with the data, they may not reach the data users who have downloaded and used the data in the past. We utilized the database of historical data download history to identify all users who have downloaded the dataset affected as part of a data reprocessing and communicate via an automated email to them a summary of changes to the data to inform them of the change. The email notifications are also sent to instrument principal investigators, and developer of Value Added Product that are using the data product and thus may be affected by the change in the data quality.

VIII. Conclusions

Data quality management and improvements for large scientific data sets like those managed by Atmospheric Radiation Measurement program pose a complex big data challenge. We described a provenance-aware computational workflow for data quality management and improvement for Petascale archive of thousands of science data streams from hundreds of instruments across the globe. End-to-end workflow enables collection of data quality issues for any data set to its processing and archival while tracking provenance information during each step of the data processing life cycle. Workflow enables the capability for symbolic equation processing that allows for instruments experts to provide quality improvement suggestions in an easy format while automating the data reprocessing pipe in a effective high performance computing environment. Provenance aware framework logs all pertinent information which are also communicated with all relevant users and stakeholders. Developed workflow would enable the fast resolution of data quality issues within ARM and provide better quality atmospheric science data to broader science users.

Acknowledgments

This research was supported by the Atmospheric Radiation Measurement (ARM) user facility, a U.S. Department of Energy (DOE) Office of Science user facility managed by the Office of Biological and Environmental Research. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725. This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid- up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

References

[1] D. D. Turner and R. G. Ellingson, "Introduction," *Meteorological Monographs*, vol. 57, pp. v–x, 2016. [Online]. Available: https://doi.org/10.1175/AMSMONOGRAPHS-D-16-0001.1

[2] R. A. Peppler, K. E. Kehoe, J. W. Monroe, A. K. Theisen, and S. T. Moore, "The ARM Data Quality Program," *Meteorological Monographs*, vol. 57, pp. 12.1–12.14, 2016. [Online]. Available: https://doi.org/10.1175/AMSMONOGRAPHS-D-15-0039.1

[3] R. A. Peppler, K. E. Kehoe, K. L. Sonntag, S. T. Moore, and K. J. Doty, "Improvements to and status of ARM Data Quality Health and Status System," *15th Conf. on Applied Climatology, Savannah, GA, Amer. Meteor. Soc.*, vol. J3.13, 2005. [Online]. Available: http://ams.confex.com/ams/pdfpapers/91618.pdf

[4] R. A. Peppler and Coauthors, "Quality Assurance of ARM Program Climate Research Facility data," *Tech. Rep. DOE/SC-ARM/TR-082*, vol. 65, 2008. [Online]. Available: http://www.arm.gov/publications/tech_reports/doe-sc-arm-tr-082.pdf

[5] R. W. Watson, "High performance storage system scalability: Architecture, implementation and experience," in *Proceedings of the 22Nd IEEE / 13th NASA Goddard Conference on Mass Storage Systems and Technologies*, ser. MSST '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 145–159.

[6] Oak Ridge Leadership Computing Facility, "High Performance Storage System (HPSS)," https://www.olcf.ornl.gov/olcf-resources/data-visualization-resources/hpss/.

[7] I. Foster, "Globus online: Accelerating and democratizing science through cloud-based services," *IEEE Internet Computing*, vol. 15, no. 3, pp. 70–73, May 2011. [Online]. Available: https://doi.org/10.1109/MIC.2011.64

[8] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, "Software as a service for data scientists," *Commun. ACM*, vol. 55, no. 2, pp. 81–88, Feb. 2012. [Online]. Available: http://doi.acm.org/10.1145/2076450.2076468

[9] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, v. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz, "Sympy: symbolic computing in python," *PeerJ*

Computer Science, vol. 3, p. e103, Jan. 2017. [Online]. Available: https://doi.org/10.7717/peerj-cs.103

[10] Dask Development Team, *Dask: Library for dynamic task scheduling*, 2016. [Online]. Available: https://dask.org

[11] M. Macduff, B. Lee, and S. Beus, "Versioning complex data," in *2014 IEEE International Congress on Big Data*, June 2014, pp. 788–791. [Online]. Available: https://doi.org/10.1109/BigData.Congress.2014.124