# *Application of Practical Methods for Data Quality Analysis at Wind Sites*

Hector Lopez
*Florida Atlantic University*
*NextEra Energy Resources*
Jupiter, FL. United States

Dr.Yufei Tang
*Florida Atlantic University*
Boca Raton, FL. United States

***Abstract*—Wind site data consumption is increasing beyond the traditional needs because of advances in artificial intelligence and machine learning. The data from wind sites can be unruly and hard to clean for data scientists. This paper tries to test out simple Data Quality rules on real world wind sites and provides practical solutions on the challenges and interesting complications that can arise when attempting to characterize the data. The end result is a set of methods that can be applied to traditional wind site data to enhance real-time data operations or machine learning on the operational data stored in the cloud or consumed for streaming into machine learning systems. The author uses references from literature to define what DQ means in the wind site domain and in particular SCADA data for wind turbines.**

***Keywords—IoT, Data Quality, Smart Grid, Wind Turbines, Cloud, Statistical Detection, Machine Learning, Infrastructure***

## I. Introduction

Wind power is the most growing renewable source, however the operation and maintenance of the wind turbines account for 25%–35% of the generation [1]. In order to increase the economic competitiveness with respect to fossil fuels and accelerate the transition towards ecologically sustainable systems, there is a need for a more efficient management and this requires better monitoring of wind turbines. The trend of operational technology such as industrial control systems has moved towards more open communication between devices. The machine to machine communication across networks has increased data transfer and data consumption by users and automated operational systems.

Traditionally wind sites have collected real-time data on locally hosted databases. The historical data would be used to trend site performance, build business models, perform audits and more. The data would be stored in aggregate form to reduce the amount of data collected, typically because of the constraints of the computer hosting the database. Often 10-minute records would suffice for any historical data analysis. Today, more data is requested from these systems to support machine learning, elastic computing, and artificial intelligence [2]. The growth of cloud storage, services and IoT technologies has decreased the dependency on hosted databases and older protocols, allowing for more granular data to be captured and processed at higher rates. [3].

The data must be trustworthy in order for any analysis or models to provide benefit. Making the data trustworthy has been investigated in recent literature through the use of data quality rules and anomaly detection. Techniques utilizing game theory [4], multi-variate state estimation, and artificial neural networks have been reviewed. [11] Each technique requires domain knowledge of the data and an understanding of the infrastructure or metrics involved in collecting the data. This leads to data quality rules to be geared towards specific domains [5].

The author attempts to apply simple data quality rules to real world infrastructure streaming data sets from various wind sites. Generic data quality metrics have been chosen as fundamental metrics for end users in the field of wind power. The data quality rules are defined broadly then they are given a practical definition considering real world constraints. Finally, a review of results and unforeseen issues is provided for the reader along with possible suggestions to expand the initial methodology for the data quality rules in order to mitigate some of these issues.

## II. Overview of Wind Site Architecture

A wind site is comprised of many wind turbines. The power of each wind turbine is controlled as one generator on the power grid at a point of interconnect (POI) with the grid. Operators utilize special software that oversees all of the complicated pieces of a wind site in order to meet the demand at the POI. The nuances of the wind site make it a challenge to generalize what the quality of the data provided by these subsystems. Practical implementations of seemingly simple rules have to be engineered to work in the environment of a wind site. An overview of the components of the infrastructure and communications currently implemented on real wind sites is provided as a primer for the steps taken to apply the data quality rules.

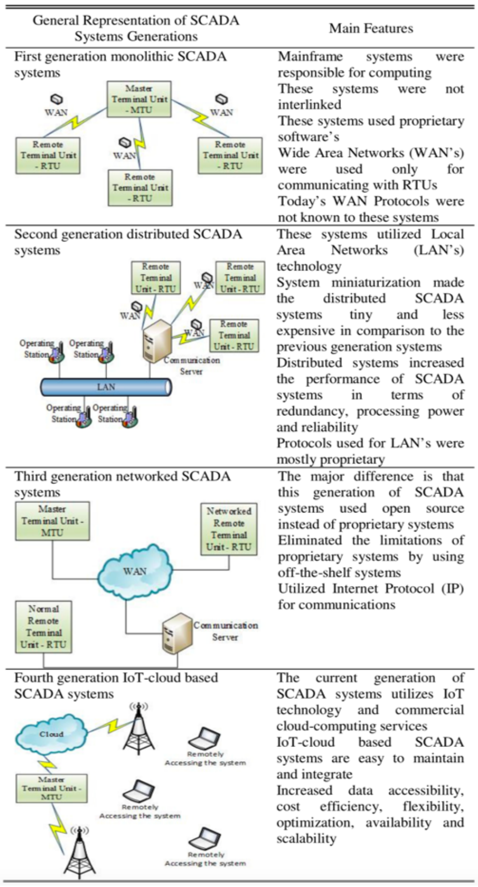
### *A. Wind Site Infrastrucure*

The wind site infrastructure can be generalized as a third generation networked SCADA system as described by Sajid [6]. A third-generation network site, as shown in Fig.1, does not have full cloud connectivity, but it is still able to transmit local data to a cloud infrastructure. A synonymous architecture would be systems that have historian software with remote server hosting and own a central datacenter.

The required data to be transmitted from the wind site to the SCADA system is divided in four categories:

- Substation data (statuses, alarms, meters, etc.)
- Weather tower / Meteorological data
- Turbine data
- Production data

The wind site communication system used at the test sites is represented in Fig. 2. A data concentrator collects the data from multiple sources. The wind turbines and weather towers are connected to the substation using fiber optic connection. The wind turbines are then connected to a proprietary server which is connected to the data concentrator or directly connected to the substation network. In addition, the data concentrator collects the information of intelligent electrical devices (IED's) such as meters, protection equipment and power quality modules. The collected data will be subsequently published to different data clients such as distribution grid SCADA, historian servers and local (Human Machine Interface) HMI. Modern wind turbines record more than 1000 variables at intervals of 50ms to 10 min. by means of their SCADA system. [1] This paper will only focus on data from the wind turbines and ignore results for secondary data sources.



| General Representation of SCADA Systems Generations | Main Features |
|---|---|
| First generation monolithic SCADA systems | Mainframe systems were responsible for computing<br>These systems were not interlinked<br>These systems used proprietary software's<br>Wide Area Networks (WAN's) were used only for communicating with RTUs<br>Today's WAN Protocols were not known to these systems |
| Second generation distributed SCADA systems | These systems utilized Local Area Networks (LAN's) technology<br>System miniaturization made the distributed SCADA systems tiny and less expensive in comparison to the previous generation systems<br>Distributed systems increased the performance of SCADA systems in terms of redundancy, processing power and reliability<br>Protocols used for LAN's were mostly proprietary |
| Third generation networked SCADA systems | The major difference is that this generation of SCADA systems used open source instead of proprietary systems<br>Eliminated the limitations of proprietary systems by using off-the-shelf systems<br>Utilized Internet Protocol (IP) for communications |
| Fourth generation IoT-cloud based SCADA systems | The current generation of SCADA systems utilizes IoT technology and commercial cloud-computing services<br>IoT-cloud based SCADA systems are easy to maintain and integrate<br>Increased data accessibility, cost efficiency, flexibility, optimization, availability and scalability |

Master Terminal Unit - MTU
WAN
Remote Terminal Unit - RTU
Operating Station
Communication Server
LAN
Networked Remote Terminal Unit - RTU
Normal Remote Terminal Unit - RTU
Cloud
Remotely Accessing the system

Fig. 1. General Representation of SCADA system generations

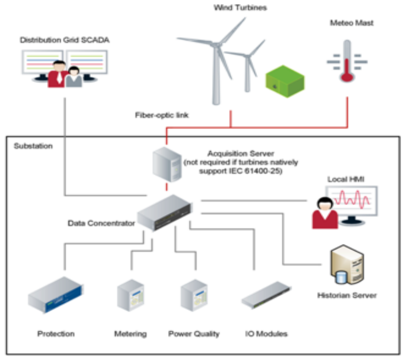


Fig. 2. Typical Windfarm communication archtictecture

### B. *Wind Site Communication*

The SCADA server provides a way for users to extract real-time data from the system in order to analyze and record the data. The protocol used to expose this data can be any of the standard Industrial Control System(ICS) protocols, such as DNP3, OPC, Modbus, etc.

The common protocols used at wind sites are Modbus and OLE for Process Control (OPC). Modbus is a register-based protocol that is open and easy to implement. Many industrial control sensors and remote terminal units (RTU's) utilize this protocol. Modbus protocol is typically seen on substation and met tower data sources. When it comes to SCADA technologies wind turbine data is typically provided to the user through the OPC protocol. This paper will focus on the OPC protocol standard. Other communication protocols such as DNP3,60870.5 will not be reviewed because of its minimal adoption at the wind sites used for testing. [7]

An OPC Connection requires an OPC client, an application that allows a user to connect to a local or remote OPC server and browse the tags. The browsable group allows settings that can affect the collection of the data. Update Rate, Percent Deadband , Async/Sync , Time Bias . By focusing on these group setting changes to the capture of the data can be affected. Intrinsic data quality measurements can suffer if these settings are not tuned properly.

## III. Data Quality Methods

Defining data quality rules can be arbitrary or specific to the end user. According to the survey's reviewed on the topic the paper proposes five dimensions of data quality that is pertinent to the domain of wind site data. [5] The dimensions are defined by describing the properties that the dimensions try to address when dealing with the dataset.

Given a time series $T = \langle t_0, t_1, \dots, t_L \rangle$ where $L$ is the length. $D = \{T_1, T_2, \dots, T_N\}$ is a collection where $N$ is the number of time series sets in the dataset. The SCADA system channels are timeseries data, $T$. The entire collection of channels would then be given as $D$.

TABLE I.

Data Quality Dimension Definitions

| ***Data Quality Dimension*** | ***Properties*** | ***Description*** |
|---|---|---|
| Connectivity | Infrastructure, Application, Site | Measure of source availability while collecting data |
| Completeness | Critical Data, Dark Data, Data Leakage | Measure of data collected from a specified data set |
| Timeliness | Processing Latency, Synchronization | Data synchronized across all sources in spite of time stamp capture or system error |
| Accuracy | Sampling Resolution, Precision, Filtering | Measure of how accurate the data captured represents the generated data (in time and value) |
| Consistency | Batching, Smooth Variance, Presentation | Measure of data collected from a specified data set over time |

## IV. Data Connectivity

### A. Methodology

The connectivity of the data is often seen as a trivial step, measuring the data source's ability to provide information when requested. Typical faults that would interfere with this capability are identified and rectified by the user. The problem occurs when the faults are intermittent, seasonal, or indicative of system degradation. Utilizing protocols that are fault aware and provide error codes can help facilitate a "Connectivity" metric for a channel of data over time. It can be aggregated to the wind turbine level and even to the wind site level.

A data connectivity metric can be used as a measure for quickly identifying data loss at a channel level, turbine, or site level. Each of these levels can be indicate different fault modes for the wind site.

A "Good" or "Bad" value can be assigned to each channel then a sliding threshold can be placed to assess how many "Bad" channels of data would constitute a "Bad" wind turbine as far as connectivity is concerned. This paper proposes utilizing a percentage for Connectivity at the turbine level that can later be leveraged for setting thresholds as the user needs. The channel data may be providing sensor data that can be seen as invalid if most of the turbine data is no longer connected. [9]. Given $D = \{T_1, T_2, \dots, T_N\}$ is a collection where $N$ is the number of channels and $T = \langle t_0, t_1, \dots, t_L \rangle$, take $T$ to indicate any sample in time as a snapshot.

$$Turbine\ Connectivity : \frac{\sum_{i=1}^{N} f(T_i)}{N} * 100 \qquad (1)$$

### B. Application

The wind sites used for applying the methodology contained OPC servers that provided the real time data from the SCADA. Each channel from the OPC server contains a quality code property that was used to identify if the channel was "Good" or "Bad". The OPC quality code is made up of 16 bits. The high 8 bits are available for vendor specific use and should be all 0's when not used. The low 8 bits are broken into three sections. The first two bits can pass the meaning Good, Bad or Uncertain. Using the OPC error codes the connectivity of the channel can be verified through an exception-based process. Note that even if the channel is available some of the errors can still occur indicating a warning or even used for predicting system degradation.

The extensive errors that can occur can be used to break down various categories of errors. In practice the complications of adding and managing new connection categories is not tenable across multiple SCADA technologies and could introduce false positives if the error properties are not properly supported during commissioning. To bypass these issues the error codes where binned into either "Good" or "Bad".

The activation function, $f(T_i)$ ,that utilizes the error codes is described as an algorithm. The condition of all the channels is resolved using the algorithm and the final activation value would result in a 1 or a 0. The average would then be calculated using the formula described in (1) for the connectivity metric percentage.

TABLE II.

**Algorithm:** GetConnectivity

```
input: opc_error_code
bitmask_good = 0x03;
if ((opc_error_code & bitmask_good) > 0)
    return 1;
else
    return 0;
```

**Algorithm:** GetTurbineConnectivity

```
input: channels
sum=0;
total= channels.length;
for(i=0; i < channels.length; i++)
  total+= GetConnectivity(channels[i].opc_error_code);
end
return ((sum/total) * 100);
```

- *Data Connectivity:* A percentage, 0% to 100%
- *Practical Definition*: Number of "Good" data channels at a given turbine over the total number of channels registered at a given turbine. Where "Good" is defined by the first two bits of the OPC quality code provided for that channel.

TABLE III.

**OPC** ***Error Code***

| ***OPC Error Code*** | ***Description*** |
|---|---|
| OPC_E_BADTYPE | The passed data type can not be accepted for this item from server |
| OPC_E_BADRIGHTS | The Item is not having either Readable or writable access rights |
| OPC_E_RANGE | The value was out of Range |
| OPC_E_INVALIDHANDLE | Clients Item handle is invalid when requested to server |
| E_NOINTERFACE | The possible version conflict between the OPC DA server version and OPC Client version while communicating |
| OPC_E_UNKNOWNITEMID | The Item ID is not part of OPC Namespace in the OPCDA server |
| OPC_E_INVALIDITEMID | The client requested item Name has invalid convention (for ex some invalid characters) |
| OPC_E_DUPLICATENAME | Trying to add a group which is already present in server |
| OPC_E_NOTSUPPORTED | If a Client attempts to write any value, quality,timestamp combination and the server does not support the requested combination(which could be a single quantity such as just timestamp), then the server will not perform any write and will return this error code |
| E_OUTOFMEMORY | Not Enough memory to complete the requested operation. This can happen any time the server needs to allocate memory to complete the requested operation |
| OPC_S_CLAMP | The Value was accepted but was clamped |
| E_INVALIDARG | An invalid argument was passed(like when client requests data to server the argument of dwcount should be >0 but if dwcount=0 then this error code will be returned |
| CONNECTION_E_CONNECTIONT | The client has not registered it communication channel with server for the data updation |
| OPC_E_DEADBANDNOTSUPPORTED | The dead band is not supported by the server |

| **OPC** ***Error Code*** | |
|---|---|
| OPC_S_UNSUPPORTABLERATE | Server does not support requested rate,server returns the rate that it can support in the revised sampling rate |
| OPC_E_NOBUFFERING | The server does not support buffering of data items that are collected at a faster rate than a group update rate |
| OPC_E_UNKNOWNPATH | The Item's access path is not known to the server |
| OPC_S_INUSE | The operation cannot be performed because the object is being referenced |

### *C. Summary*

Connectivity can be a simple measure of a prebuilt indicator for each channel at the wind site. Several GE wind sites where presented as using OPC for the SCADA data. The OPC protocol was explored and utilized to build a method of calculating the connectivity of a single channel. The single channel values where aggregated to eventually provide a connectivity metric at the turbine level.

## V. Data Completeness

### *A. Methodology*

The completeness metric can be used to verify that all the channels that should exist from a wind turbine are available at the turbine. This metric is meant to provide the end user with a catalog of channels they are able to access from the wind turbine. The standard provided by the IEC 61400-25 attempts to provide a hierarchy of logical nodes that can describe the segments of a typical wind turbine. [8] The standardization of wind turbine naming topology can help create a catalog of channels of data the end user would ultimately need.

The method to provide a "Completeness" metric can begin by having the end-users commit to a "critical" tag list. The logical nodes that where most critical to the users where identified using the IEC standard naming. The critical tags can now be used to create a percentage of what would be available at the wind turbine level. Similar to the connectivity metric this completeness metric can be aggregated to the wind site level.

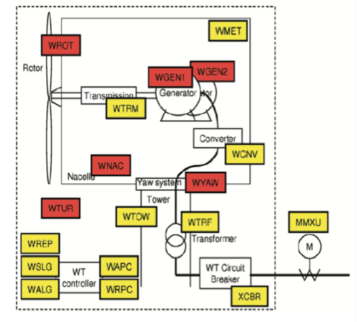


Fig. 3. IEC 61400-25 Logical Nodes for Wind Turbine Components

### *B. Application & Results*

A script was created that would take a list of "critical tags" and then try to identify if these critical tags were available on a particular wind turbine. The percentage was used to create a site average percentage. There where complications when implementing this methodology at ten different wind sites with GE technology wind turbines.

- *Data Completeness:* A percentage, 0% to 100%
- *Practical Definition*: Number of user-defined critical channels of data that are available at a wind turbine over the total number of critical channels.

The report brought to light many challenges in trying to create a Data Completeness metric for disparate systems, and even for similar systems.

- IEC Standard is not adopted by many SCADA manufacturers. Definition of the channels from the critical tag list required researching OEM manuals; SCADA Versions: WindSCADA 10.0,11.0, 12.0, 14.0 & Controller Types: MarkVI, Bachmann.
- Wind turbine software was not same across wind turbines at the same wind site, leading to different and unknown channels that are missing.

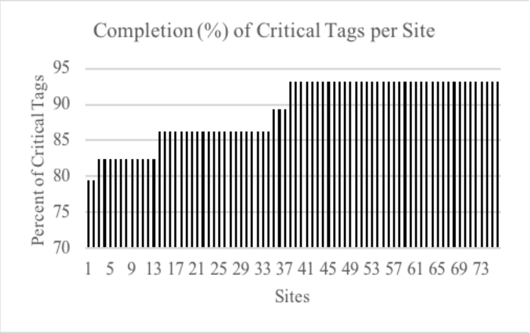


Fig. 4. Completeness metric acros multiple sites that supported critical tags requested by user.

TABLE IV.

| **Wind Site Data Completeness Metric Report** | | | |
|---|---|---|---|
| ***Site*** | ***Number of Turbines*** | ***Total Number of SCADA Tags*** | ***Wind Site Data Completeness*** |
| #1 | 37 | 8030 | 79% |
| #2 | 87 | 4618 | 79% |
| #3 | 72 | 5408 | 82% |
| #4 | 26 | 1152 | 82% |
| #5 | 45 | 86800 | 82% |
| … | … | … | … |
| #76 | 33 | 26680 | 93% |

### *C. Summary*

In conclusion the report was useful to identify gaps in SCADA configuration and Wind Turbine software. The fact that there were no sites that provided a 100% matching of the critical tags indicated a gap in the request by the user for a channel of data that may not exist or is misidentified. This is important and usable in anomaly detection because during normal operations the wind turbine controller software can be upgraded or modified. When the controller software is changed the channels available to the user on the SCADA would also change. When machine learning models are replicated to other systems an audit of the available channels can be used to see if the assumed features still exist for the model to be successful [11].

## VI. Data Timeliness

### A. Methodology

The Data Timelines metric is meant to track any statistically significant drift in the length of time or a change to the reported timestamp of a channel of data. All of the data collected at the wind site can be read from the single SCADA software as the source of the data. Since this is ultimately the same as reading all of the data from a single sensor, the possibility of the entire SCADA system or logging software to introduce delays is possible. A method of increasing robustness would be to introduce more sensors such as a secondary source for the same measurement. [9]

Given multiple channels of timeseries data $T_1 = \langle t_0, t_1, \ldots, t_L \rangle$, and $T_2 = \langle t_0, t_1, \ldots, t_L \rangle$ to represent sensor readings from two different sources. To track the drift between these two sensors a deviation metric can be created over a given time window. Timeliness would then be a metric of how much drift occurs for a given channel as a deviation metric then averaging it across all channels for a single turbine.

### B. Application

Wind sites will typically have a power meter that measures the output of the site. This power meter allows the SCADA software to control the output by regulating the setpoint of each wind turbine. This Wind Farm Management System (WFMS), can accurately determine what the site output should be. There are also power meters at each of the wind turbines that relay the power data to the SCADA software. The SCADA software is also providing the site power measurement to the user. There could be a latency between the power reading from the SCADA software and the power reading from the power meter. The power meter is usually calibrated to provide an accurate measurement, so if the two readings begin to drift from each other it would be indicative of power meter calibration error or an artificial latency affecting the channel from the SCADA software.

The report shows a comparison of the SCADA active power reading and the sites meter active power reading, using a statistical sampling method.

- *Data Timeliness:* A percentage, 0% to 100%
- *Practical Definition*: Percent deviation from a running average created by taking the difference between two channels of the same data (such as site active power) that come from different sources.

The power metric from the wind site was chosen at the meter and compared to the power provided by the SCADA. A deviation metric was taken over several days. The two channels where measured across different events. A disconnected event shows the deviation between the signals. The deviation between the signals grows over time. The deviation can be tracked as a rate of change. (2)

$$\frac{\Delta\left(\frac{|T_1 - T_2|}{L}\right)}{\Delta t} \quad (2)$$

TABLE V.

| Wind Site Data Completeness Metric Report | | |
|---|---|---|
| **Average** | **Count** | **Standard Deviation** |
| -16.662278 | 2810 | 23.2856983 |

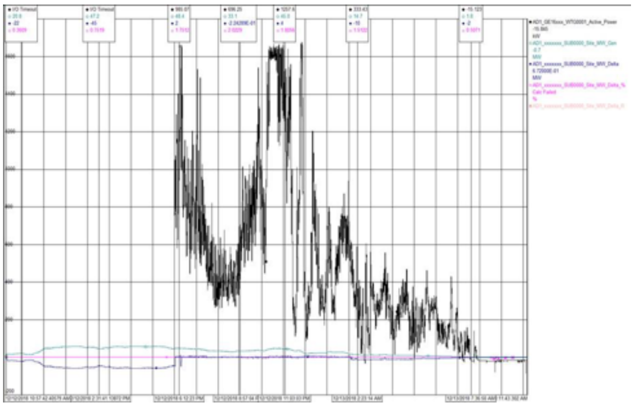


Fig. 5. Deviation of site meter MW and substation site MW

### C. Summary

The data quality metric for timeliness should focus on detecting any deviations between the data source from other sensors. The power meter can be used as a benchmark to detect deviation.

## VII. Data Accuracy

### A. Methodology

The Accuracy in the data involves two major components, the Resolution of the data that deals with the time dimension and the Precision of the data that deals with the space dimension. The resolution can be measured by performing an analysis over many snapshots at an arbitrary window size. Choosing a window size of 1 minute helps to standardize the unit of measure when discussing resolution. As discussed in other sections of the paper the limitation of data points is usually the update rate that limits changes of a value typically to 1 second. This change rate would make it reasonable to measure resolution in minutes. Resolution metric must be identified statistically across many sites by benchmarking against high frequency channels/tags of a high-res site.

*1) Implement a "High Frequency" Channel :* A high frequency channel is a channel from the SCADA system that is constantly changing close to the limit of the transfer rate for the system. A downstream timestamp can be used to measure changes per second. This measurement can fluctuate based on infrastructure limitations such as protocol packet transmission times. It would only measure a single channel and not provide insights to resolution of other sensors effected by infrastructure downstream of the collection point.

*2) Benchmarking with a "High Frequency" Channel*: Choose top channels with highest frequency for benchmarking the capability of all channels for the SCADA software. Continuous capture of the data would be ideal but in practice it becomes very time consuming and resource intensive when the data becomes very large. A statistical approach. Take $n = 100$ discrete uniformly distributed values of the resolution metric (points/min) of a high frequency channel over the course of 24 hours (1440 min.). Let $R_i$ be the resolution sample taken at a minute, for a given channel, $R$.

$$\bar{R} = \frac{\sum_{i=1}^{n} R_i}{n} \ , \sigma = \sqrt{\frac{\sum_{i=1}^{n} (R_i - \bar{R})^2}{n}} \ , \ i \sim U\{1440\} \qquad (3)$$

### B. Application

Set a tolerance for the channel frequency benchmark and compare deviation from norm when grading similar channel/tag at other sites.

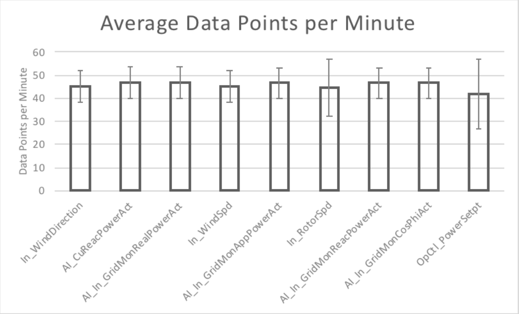


Fig. 6. Average data points per minute, showing resolution of critical SCADA tags averaged over all wind turbines at a running wind farm

### C. Summary

The wind sites in practice did not have a channel that could be used to measure the performance of the SCADA in providing consistent high-res data. The channel that changes at a sub second level as a constant timer is needed for this type of metric. The approach that was taken ,was to take historical performance of a top channels that seem to be changing at high frequency and use it to measure the SCADA's resolution performance as a whole. The method is not as inclusive or precise but can be sufficient given enough samples and statistical variance when selecting the data. The average resolution can be used for the DQ Accuracy metric as a primary indicator. Consideration should be taken into using methods that also vet the precision of the data using data type discovery.

## VIII. Data Consistency

### A. Methodology

The consistency of the data is similar to server uptime metrics. Where the data quality metric would be most effective should be considered. Measuring SCADA server uptime is not ideal in this case because it is only looking at the hardware performance [12]. An indicator that is closer to the end users' consumption needs should be considered. The proposed method is to take the measurement at the source of record for the data. By analyzing the data contextually over an arbitrary period of time the data can be provided an "uptime" metric as a percentage. This metric can be used as the primary indicator for Data Consistency.

A secondary indicator would be if there is unwanted sparseness in the data set. These sparse data behaviors can be corrected through generating synthetic time series to augment the data [13]. Gaps in the data set can also be measured through data quality metrics like connectivity and accuracy. The sparseness of a data set can be analyzed in many ways so the consistency metric will be focused on just the ability to capture the data stream consistently over a period of time.

### B. Application & Results

A "health" tag was artificially generated and programmed to deliver a value every minute to the database. A simple query of all health tag data would result in a count over the time window of 30 days. The count could be considered uptime in minutes. The uptime in minutes was divided by the 30 days to create an uptime percentage. This process reduced processing time and was an efficient way to calculate uptime across many sites.

The process ran as a script that queried the database and calculated the uptime percentage for 100 different wind sites. The report accurately showed sites that were not available during the 30-day period as 0%. The report also revealed a crucial gap in performance indicating that data records for the channel were missing at least 20% of the time on average. (See Fig.5)

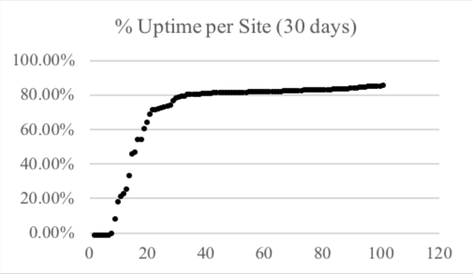


Fig. 7. Uptime percentage over the course of 30 days at a 1-minute resolution. Measured by historical data across 100 different wind sites.

After investigation of the uptime performance the historical storage system used was diagnosed to be under large amounts of strain. The database was reporting multiple system errors that prevented the writing of messages into the database over the same time period the report was taken.

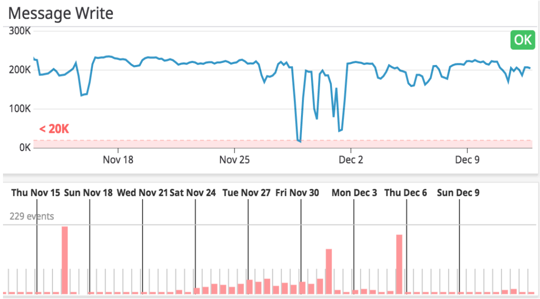


Fig. 8. Metric indicating the writes to the database that stores all messages from all windsites (top). System error events occurring, indicative of write failures and possible lost data resulting in impact to uptime (bottom).

### C. Summary

In conclusion the consistency of the data can be primarily measured by uptime. The application of a tag that can provide 1-minute data was helpful in identifying an issue resulting in loss of data. Using a "ideal" value that is recorded at a consistent frequency allows for an efficient query to pull back the uptime metric across a large set of sites for a large period of time.

Further considerations should be considered when trying to provide a Data Quality metric for Consistency because the uptime metric does not account for data sparsity.

## IX. Conclusion

The paper has reviewed typical infrastructure of a wind site and has focused on wind site that are collecting more data into remote systems like the Cloud. These new data sets can be larger than traditional operational data and require cleansing by subject matter experts. The data cleansing can be assisted by deigning data quality rules that vet the incoming data and assist data scientists, and other end users in utilizing the large data sets. The author focuses on five different forms of data quality, known as dimensions.

Connectivity was a DQ dimension that was measured by using existing codes from the typical protocols used at the site. Other methods to aggregate the connectivity was also discussed in order to view the metric at a turbine or even a site level.

Completeness was defined by utilizing the requests of the end-user and leveraging the IEC standard that categorized critical components of the wind turbines. These critical channels where used as a list of information that can be measured by reports. The paper reviewed an extensive report across seventy different wind sites measuring how the wind sites met the user requests. Caveats and challenges where found in the nature of the SCADA software and the Turbine controller software. Proving that this is not at simple as it may seem at first glance.

Accuracy of the data can be multi-faceted but the metric was built around the resolution of the data captured. An efficient method would have been to utilize a channel that is constantly moving at a the highest resolution possible, but in practice these signals are not available. The approach taken was to benchmark performance of each site over time and track deviation in that performance.

Timeliness is a vauge term but was used to determine any drift or deviation of a signal due to infrastructure or software noise in the form of time delays. The approach taken was to watch the deviation of two channels tied to the same sensor measurement and determine if there is a phase shift of the data. The perecent change of the deviation would be the timeliness metric for the channel. It would then be able to be aggreagated across all channels for that turbine to determine the drift in the overall data for the turbine.

Finally, consistency was analyzed as a measurement of uptime similar to how server's are measured. Utilizing heart beat values that are published every minute, the uptime measurement led to a direct indicator of data recording issues.

All of these data quality rules have been proven on real wind sites and are baked into the data collection pipelines that support the delivery of data to the end-users. These DQ metrics are ways to improve the understanding of the data but can also be used to augment and correct the data that can be considered “Bad’ quality. These efforts are to be explored and implemented practically to vet the feasibility.

## Acknowledgment

Substantial help was provided by teams in the wind renewable industry at NextEra Energy. Professor Dr.Yufei Tang, was critical in designing the scope and setting the tone of the research. A great benefit came from the experience of Dr. Hooman,Yousefizadeh a F.A.U alumni and data scientist that helped in translating the anomaly detection methods he pioneered into useful methods for data quality rules.

## References

[1] Kim Schumachera , Zhuoxiang Yangb , “The determinants of wind energy growth in the United States: Drivers and barriers to state-level development”, Renewable and Sustainable Energy Reviews 97 (2018) 1–13

[2] Sunaina Sulthana Sk, Geethika Thatiparthi, Raghavendra S Gunturi “Cloud and Intelligent Based SCADA Technology”,International Journal of Computer Science and Electronics Engineering, Volume 2, Issue 3 , March 2013.

[3] Shyam R, Bharathi Ganesh HB, Sachin Kumar S,Prabaharan Poornachandran, Soman K P, “Apache Spark a Big Data Analytics Platform for Smart Grid “, SMART GRID Technologies, August 6-8, 2015, 2212-0173 © 2015 Published by Elsevier Ltd.

[4] Elisa Bertino and Hyo-Sang Lim , “Assuring Data Trustworthiness - Concepts and Research Challenges⋆” Department of Computer Science, Purdue University, USA, {bertino,hslim}@cs.purdue.edu, W. Jonker and M. Petkovi ́c (Eds.): SDM 2010, LNCS 6358, pp. 1–12, 2010. Springer-Verlag Berlin Heidelberg 2010.

[5] Aimad Karkouch, “Data Quality in Internet of Things: A state of the art survey” 2016

[6] Anam Sajid, Haider Abbas, and Kashif Saleem, “Cloud-Assisted IoT-Based SCADA Systems Security: A Review of the State of the Art and Future Challenges”, February 21, 2016, accepted March 25, 2016, date of publication March 31, 2016, date of current version April 21, 2016. Digital Object Identifier 10.1109/ACCESS.2016.2549047

[7] A. Jha, D. Dolan, T. Gur, S. Soyoz, and C. Alpdogan , “Comparison of API & IEC Standards for Offshore Wind Turbine Applications in the U.S. Atlantic Ocean: Phase II” March 9, 2009 – September 9, 2009 MMI Engineering Houston, Texas NREL Technical Monitor: Walt Musial

[8] David Yates, Jim Kurose,Prashant Shenoy, “Data Quality and Query Cost in Wireless Sensor Networks”, Proceedings of the Fifth Annual lEEE International Conference on Pervasive Computing and CommunicationsWorkshops (PerComW'07) 2007 IEEE

[9] Ting-Han Lin a , Shun-Chi Wu a,b, "Sensor fault detection, isolation and reconstruction in nuclear power plants Ting-Han Lin a , Shun-Chi Wu”. Annals of Nuclear Energy. Volume 126, April 2019, Pages 398-409

[10] Noam Erez, Avishai Wool, “Control variable classification, modeling and anomaly detection in Modbus/TCP SCADA systems” School of Electrical Engineering, Tel Aviv University, Ramat Aviv 69978, Israel, international journal of critical infrastructure protection 10 (2015) 59–70

[11] Gross, K., Singer, R., Wegerich, S., Herzog, J., VanAlstine, R., Bockhorst, F., 1997. Application of a model-based fault detection system to nuclear plant signals. In: Proc. 9th Intnl. Conf. On Intelligent Systems Applications to Power Systems, 66– 70

[12] A.H. Beitelmal, D. Fabris ,“Servers and data centers energy performance metrics”, Energy and Buildings 80 (2014) 562–569

[13] Germain Forestier, Franc ̧ois Petitjean, “Generating synthetic time series to augment sparse datasets”, Computer Science and Engineering Dpt, University of California, Riverside, USA cs.ucr.edu, 2017 IEEE International Conference on Data Mining, DOI 10.1109/ICDM.2017.106