*Methods of Data Quality Analysis and Detection for OPC Based Wind Farm SCADA Systems*

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

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

*NextEra Energy Resources*Jupiter, FL. United States

*Abstract*—In the Internet of things (IoT), data gathered from a global-scale deployment of smart-things are the base for making intelligent decisions and providing services. This paper focuses on the IoT data from a Wind Site with many Wind Turbines and aims to implement a method for detecting and tracking data quality (DQ) of the IoT data. The paper aims to identify the definitions of the DQ dimensions specific to the domain for Wind Sites. After generating a method to detect data quality a dataset from a live wind site is used to determine the effectiveness of the DQ strategy. Data capture settings are adjusted to understand the tradeoffs between DQ dimensions. Data is also categorized and analyzed in batches to understand the DQ of the captured dataset. The results explore any improvements or actionable information that can be used to improve the DQ. In conclusion the possibility of future research and enhancements is presented.

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

# 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 costs (Milborrow (2003)). [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. Modern wind turbines record more than 1000 variables at intervals of 50ms to 10 mins by means of their SCADA (Supervisory Control and Data Acquisition) system.

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. Today, more data is requested from these systems to support or implement technologies such as Artificial Intelligence (AI) [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. The amounts of varying and variable sensor data collected can be categorized as “Big Data”. The data collected is analyzed to reveal the knowledge of unseen patterns as described by Shyam [3].

In order for analysts and decision makers to produce accurate analysis, make effective decisions and take actions, data must be trustworthy. It is thus important to provide a comprehensive solution for assessing and assuring the trustworthiness of the information collected. In general, the problem of providing “good” data to users and applications is an inherently difficult problem which often depends on the application and data semantics as well as on the current context and situation. In many cases, it is crucial to provide users and applications not only with the needed data, but with also an evaluation indicating how much the data can be trusted. Bertino and Lim [4] proposed computational methodologies to generate trust scores of every element of data in the context of distributed systems. They attempt to create trustworthiness scores for analytics. The methodologies proposed do not directly translate to the domain of wind power. A survey on Data Quality (DQ) by Aimad [5] describes the need for domain specific metrics to address the user requirements more precisely since DQ can be subjective if not framed in the correct context. The proposed. The author uses large datasets collected from several wind sites to identify methods to implement data quality rules. A group of similar GE WindSCADA sites provide real-world problems to basic DQ metric implementations shedding light on what are some practical steps to establish automated methods when moving from traditional SCADA data collection rates to larger datasets meant to be used by AI and cloud computing.

# Overview of Wind Site Architecture

The author has prioritized several metrics according to operational use cases for wind sites. As previously mentioned a collection of wind sites has been used to verify generic methods of collecting DQ metrics. These wind sites utilize the GE WindSCADA software for its site SCADA. 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, instead it is still able to transmit local data to a cloud infrastructure through increasing network layers of security; device, control and business, where the cloud data is stored on the business network.

1. General Representation of SCADA system generations

## Wind Site Communications

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 farm 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. This paper focuses on the data published to the cloud from the local historian client. The architecture of the site was defined from the site infrastructure drawings but similar systems are referenced in the literature as proposed by Sajid [6].

1. Typical Windfarm communication archtictecture

## SCADA OPC Protocol

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 protocol that shall be discussed in this paper is OPC. Data is gathered via an (OPC) OLE for Process Control [7] with update periods of 5–10min, producing several types of indicators. Only failure events and statistical indicators are kept. 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.

# Defining Data Quality Dimensions

When collecting large datasets from wind sites, rules

need to be defined in order to describe the quality of the data before its used in modeling and real time operations. Defining these rules can be arbitrary or specific to the end user. A set of general methods can be proposed if the domain and end-user is known. [5] This paper will focus on how to measure connectivity of the SCADA data, how to measure completeness of the data set for each wind site, propose a method for measuring timeliness of the data and finally investigate the problems of representing accuracy of the data. All of the dimensions mentioned will have a method that can be directly applied to real wind sites and even used for anomaly detection.

Data quality dimensions are arbitrarily defined in Table I. and broken up into classes for the reader to see metrics that are intrinsically configured by settings of the capturing mechanism and contextually measured using historical trends. The dimensions also fall under different properties and are specifically described with the domain of a wind site architecture in mind. [5]

### Intrinsic Data Quality

Intrinsic DQ dimensions can be directly measured from data collection system.

### Contextual Data Quality

Contextual DQ Dimensions require insights, benchmarks, analysis and historical data to measure.

| Data Quality Dimension Definitions | | | |
| --- | --- | --- | --- |
| Data Quality Dimension | Properties | Description | Wind Site Data Definition |
| Connectivity | Infrastructure, Application, Site Intrinsic | Measure of source availability while collecting data | Defined as ‘Bad’ if data is obstructed, otherwise, ‘Good’ |
| Completeness | Critical Data, Dark Data, Data Leakage Intrinsic, Contextual | Measure of data collected from a specified data set | The percentage of channels that exist at a wind turbine that are required by a customer |
| Timeliness | Processing Latency, Synchronization Intrinsic | Data synchronized across all sources in spite of time stamp capture or system error | Latency of real power, measured in milli-seconds between SCADA channels and site power meters |
| Accuracy | Sampling Resolution, Precision, Filtering Intrinsic, Contextual | Measure of how accurate the data captured represents the generated data (in time and value) | high frequency channels at a wind turbine measure the percent deviation between the historical resolution and the sampled average resolution. |
| Consistency | Batching, Smooth Variance, PresentationContextua | Measure of data collected from a specified data set over time | Uptime percentage measured in number of minutes the system is operational over the course of a year. |

## Data Connectivity

The connectivity of the data can be understood in the OPC protocol as a valid connection to collect data. With this quality dimension the focus is on keeping track of how the data is not connected at any given point in time. By leveraging OPC quality codes one can begin to track the possible failure modes of each of the items.

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. See Table II.

| 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 |
| E\_FAIL |  |
| 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\_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 |

Running a report across critical channels at several wind sites provided an overview of tag connectivity. The report was scripted to run continuously across multiple sites to cover different site conditions.

The connectivity report looks at the error code of each tag across the time window and sums the time the tag was in error. It then sums that number across the entire wind turbine and then again across the entire site.

| Data Connectivity Analysis | | | |
| --- | --- | --- | --- |
|  | ***Good*** | ***Bad*** | ***Unknown*** |
| True |  |  |  |
| False |  |  |  |

In conclusion the OPC protocol inherently provides an effective method for detecting connectivity issues down to the individual tag level. Using these error code as an aggregate does not rule out false positives. In order to effectively measure the connectivity an additional “Heart Beat” signal was created that would increment every second in order to rule out conditions where the data is artificially reporting as “Good”.

## Data Completeness

Define what a critical set of tags out of a full set of tags would be defined as completeness. The more tags the more data is ingested. This is a dimension that would be impacted if the data ingestion needs to be decreased. Completeness can be heavily impacted by connectivity. The difference between connectivity and completeness is that completeness is heavily dependent on the needs of the user. Different customers may ask for differing requests.

1. IEC 61400-25 Logical Nodes for Wind Turbine Components

The report for data completeness focused on a set of critical tags identified by converting the tags to the IEC 61400-25 standard using the GE manual. The tags where then marked as critical if they existed in the operational hierarchy of the wind turbine [8]. The report was able to effectively show the missing tags that were missing from 10 wind sites the report ran on. The report was shed light on several misconceptions, that led to poor completeness metric:

* Wind turbine software is not always the same across turbines of similar technology, leading to different and unknown channels at a site
* Sites may not always have the same SCADA configuration i.e. sites #8, #9, #10 did not have the same OPC server naming convention as the others. (see Table III)

In conclusion the report was useful to identify gaps in SCADA configuration and Wind Turbine software. This is important and usable as a metric for anomaly detection because during normal operations the wind turbine can be upgraded or modified potentially affecting machine learning systems that are dependent on particular channels of data.

| Completeness Metric Report | | | |
| --- | --- | --- | --- |
| ***Site*** | ***Number of Turbines*** | ***Total Number of SCADA Tags*** | ***% Completeness*** |
| #1 | 45 | 137651 | 81% |
| #2 | 60 | 442212 | 81% |
| #3 | 57 | 172915 | 78% |
| #4 | 142 | 321282 | 78% |
| #5 | 133 | 428756 | 78% |
| #6 | 125 | 329252 | 75% |
| #7 | 64 | 126718 | 68% |
| #8 | 80 | 163245 | 46% |
| #9 | 34 | 24216 | 46% |
| #10 | 10 | 36196 | 42% |

## Data Timeliness

Define Timelines as the latency across an ingestion, storage and consumption pipeline that occurs due to the mechanisms of the pipeline. The timeliness quality is not just latency but in a SCADA environment it also tracks the time stamp of when the data was captured, received recorded and consumed. These varying timestamps must be synchronized over a common time server. The challenges of having segregated networks is that there is no common time sync . Manytime the timestamps will become out of sync causing impact to the validation of the origin time.

## Data Accuracy

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.
* Take 100 discrete uniformly distributed values of the resolution metric (points/min) of a high frequency channel over the course of 24 hours.
* After characterizing a channel compare it to other turbines at the site to get a site level frequency range average with standard deviation.
* Set a tolerance for the channel frequency benchmark and compare deviation from norm when grading similar channel/tag at other sites.
* Choose top 10 channels with highest frequency for benchmarking.

By analyzing multiple windows over another arbitrary period of time and averaging over the samples allows one to view what the resolution is on average for a given tag. A window of 1 minute was chosen with 49 windows taken over the course of 48 hours. The windows where taken periodically with an hour apart. The resolution metrics of each window was averaged for each tag. The standard deviation was used to create a Margin of error with a confidence metric of 80%.

1. Average data points per minute captured 49 times over the course of 48 hours at the top of each hour, showing resolution of critical SCADA tags for an arbitrary wind turbine at a running wind farm

The OPC DA server replies with the selected rate, which is the closest multiple of the Maximum Client Rate configured in the WorkstationST OPC DA server tab, or in the Runtime Monitor Config Options menu. For example, if the Maximum Client Rate is set to 100 ms and the client requests a rate of 80 ms, the client is given a rate of 100. If the client requests 160 ms, the client is given a rate of 200 ms. The actual update rate of the variables in a group depends on the rate that the variable is being updated to the OPC DA server. EGD variables are updated at the EGD exchange rate. For SDI variables, the SDI live list is requested to the controller at the group rate. In server performance testing.

* 5000 Boolean variables changed at 640 ms, and updated on one EGD exchange at 1000 ms
* 10000 floating point variables changed at 32 ms, and updated on 40 EGD exchanges at 1000 ms
* 100 floating point variables changed at 32 ms, and updated on one EGD exchange at 100 ms

The server maximum client connection rate was set to 10 ms and one client with one group was connected with a rate of 100 ms. With the client connected, the OPC DA server used between 20 and 30 percent of a Pentium® 4 2.6 GHz CPU. Without the client connected, the CPU utilization was around 10 percent.

| Wind Farm Information | | | |
| --- | --- | --- | --- |
| ***# WTGs*** | ***Site MW*** | ***Base Rating*** | ***Rotor*** |
| 37 | 59.9 | 1.62 | 100 |
| ***Blade Types*** | ***Pitch Type*** | ***Tower Height*** | ***Controller*** |
| Glass | Salem | 80 | MarkVI |
| ***Converter Type*** | ***Software Version*** | ***SCADA Version*** | ***Converter Type*** |
| GE ESS | 44.76.00C | WindSCADA 10.0 SP2 | GE ESS |

## Equations

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Number equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop. To make your equations more compact, you may use the solidus ( / ), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in:

*a**b* 

Note that the equation is centered using a center tab stop. Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(1)”, not “Eq. (1)” or “equation (1)”, except at the beginning of a sentence: “Equation (1) is . . .”

##### Acknowledgment

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

1. Wind power is the most growing renewable source, however the operation and maintenance of the wind turbines account for 25%–35% of the generation costs (Milborrow (2003)). [1]
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

1. 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
2. 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
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.