Viharos, Zs. J.; Sidló, Cs. I.; Benczúr, A. A.; Csempesz, J.; Kis, K. B.; Petrás, I.; Garzó, A.: ”Big Data” Initiative as an IT Solution for Improved Operation and Maintenance of Wind Turbines, European Wind Energy Association (EWEA) Conference, “Make your vision reality”, 4-7. February, 2013, Vienna, Austria, S9.3, pp. 184-188.

”Big Data” Initiative as an IT Solution for Improved Operation and Maintenance of Wind Turbines

Zsolt János Viharos, Csaba István Sidló, András A. Benczúr, János Csempesz, Krisztián Balázs Kis, István Petrás, András Garzó

Computer and Automation Research Institute of the Hungarian Academy of Sciences (MTA SZTAKI), Laboratory on Engineering and Management Intelligence and Informatics Laboratory

{zsolt.viharos, csaba.sidlo, andras.benczur, janos.csempesz, krisztian.kis, istvan.petras, andras.garzo}@sztaki.mta.hu

Abstract: “Big Data” (BD) problems require handling extremely large or complex datasets that would be difficult and expensive using traditional relational databases. Software solutions with distributed processing, weakened consistency requirements and well-designed data models help overcoming scalability issues.

Wind energy systems produce extremely large datasets. Today’s wind farm operators either do not collect all available data in a central, easy to access database, or they delete valuable data, because of scalability issues of traditional databases. Emerging “Big Data” tools and algorithms enable collecting all of the most detailed data; moreover, data may not be deleted at all. This is a huge advantage for wind farm operators, because detailed information can be (re)used later for many purposes: e.g., building failure detection and prognosis models, ad-hoc analysis of the past becomes feasible.

The paper shows how detailed operation data from a large number of wind farms can be collected and stored for further use. A Business Intelligence (BI) reporting prototype system for wind farm analytics is described, with describing typical cases of wind turbine operations where advantages of the “Big Data” initiative can be exploited. Performance tests for collecting and storing of SCADA data from many wind farms prove the advantages and applicability of the proposed method.

Keywords: wind farms, operation and maintenance, big data, business intelligence

1. INTRODUCTION

Wind energy is one of the most promising branches of the energy sector [3]. A characteristic trend in Europe over the last few years is that the speed of increase in the establishment of new wind farms decreased, consequently, the importance of the wind farm operation receive more attention [12]. This trend predicts extensive use of Wind Turbine Generator (WTG) condition monitoring and supervision solutions.

Various wind turbine condition monitoring and sensing techniques related to different WTG components are compared in [6, 13]. In [14] the relation of condition monitoring to reliability calculation is explained, as a tool for handling the wearing outside of a bath curve of reliability. Combinations of turbine components and monitoring techniques are highlighted in [14], while hybrid statistical models are introduced in [11] for special problem domains.

Big Data problems require handling extremely large data sets and running complex algorithms, which are beyond the ability of commonly used software tools [24]. These problems appear more and more frequently in practice, and new software solutions are developed to solve them. According to Deloitte, by the end of 2012 more than 90 percent of the Fortune500 companies will likely have some Big Data initiatives, at least as pilot projects [19].

Originally driven by Web and telecommunication companies, a wide range of Big Data tools have arisen and matured in the last decade. They have proven to be useful for data scientists of social networks, consumer behaviour, retail, mobility and sensor data, where traditional data warehouses reached their limits. These solutions are easy to extend by predictable costs, as data sets grow. They offer value to the energy sector: they support efficient operation and maintenance by scalable wind data analytics.

In the following a prototype wind turbine business intelligence system is presented, with experiments showing how effectively Big Data tools can aid preparing large wind turbine sensor datasets for analytics. Performance tests for collecting and storing of SCADA data from many wind farms prove the advantages and applicability of the proposed method.

2. WIND FARM OPERATIONAL DATA COLLECTION AND STORAGE

Wind turbine generators are data intensive devices incorporating various sensors, as in manufacturing branch, too [17]. The sensors allow real-time condition monitoring and supervision, and enable preparation of statistical reliability models [5,4]. Wind turbines are operating in a rapidly changing and sometimes extreme environment [15]. Handling the variety of environmental effects is a hard task requiring advanced statistical and Artificial Intelligence (AI) technique based analysis. Data collection under various conditions is a necessity, and sophisticated data processing techniques are needed [10,16,18].

Current wind turbine data processing systems contain the following typical elements:

- *Sensors* of various physical-electrical effects are the initial data sources.
- *PLC(s) (Programmable Logic Controller(s))* receive information from sensors transforming electrical signals to digital data. Moreover they do many transformations and interventions into the working behaviour of wind turbine.
- *SCADA (Supervisory Control and Data Acquisition)* systems are physically connected to the sensors/PLCs collecting signal and other data.
- *Condition Monitoring (CM)* systems (beyond the PLC and SCADA components) also collect relevant signal data and produce relatively high frequency data series.
- *Industrial computers* provide local storage inside the turbine with basic calculation functions.
- *Data transmission systems* are mobile devices providing a communication channel for data transmission between turbines and data centres.
- *Data centres* collect, store and archive data of individual turbines.
- *Functional servers* typically receive data from data centres and support various data Extraction, Transformation and Load (ETL), reporting and analysis tasks.
- *Client computers* are connected to functional servers, and provide end-user interfaces.

At each step of the data chain, the amount of the data is reduced, resulting in much less information for the end-user than originally. “Big data” solutions can be applied to data centres and functional servers to decrease information loss with a high throughput, and to provide more complex analytical services.

3. THE CONCEPT OF “BIG DATA”

In case of “*Big data”* problems the size of the data itself becomes part of the problem: data becomes large enough that it cannot be processed using conventional methods [2]. Processing Web- or sensor data are considered as traditional “big data” problems.

Several new solutions arose in the last decade for big data problems. They include new generation SQL databases with massive parallelization and in-memory processing; distributed NoSQL tools relaxing the strict consistency criteria and the data model of traditional databases; and parallel data streaming frameworks providing real-time processing.

Figure 1 enumerates some big data solution providers [15]. The planes build on each other: the first plane contains infrastructure providers; the second one contains frameworks for data processing, and the third one covers analytical applications.

3.1 Tools for “Big Data” (BD)

The idea behind all of the BD tools is to “divide and conquer”, to apply distributed computing on easily extendable shared-nothing architectures. The main goal is to keep processing time linear in the input size; this way the required computing capacity remains predictable. For example, if the input data doubles, then only the number of computer nodes has to be doubled.

BD tools fall into two main categories. Data-streaming frameworks offer large throughput real-time processing, while NoSQL solutions provide persistent storage with fast data access. One of the most mature streaming tools is Twitter’s Strom [20]. The scalable and flexible model of Strom might be used for real-time, low-latency alerting, based on detailed sensor data.

We concentrate on NoSQL solutions in this paper, as alternatives of traditional business intelligence and data warehousing tools. “NoSQL” refers to non-, or “not only” relational (SQL) distributed databases. A

great variety of NoSQL tools are available today, addressing different tasks and requirements [24].

Brewer's CAP theorem [23] states that for a distributed computer system consistency (C), availability (A) and partition tolerance (P) cannot be satisfied at the same time – only two of them can. One way to categorise NoSQL tools is to specify what they give up. Traditional distributed databases do not tolerate if a partition falls out (CA). Others, for example HBase and MongoDB, might not response to a request if there is no consensus between partitions (CP). While for example Dynamo, CouchDB are always available, but they might give us false, outdated results (AP).

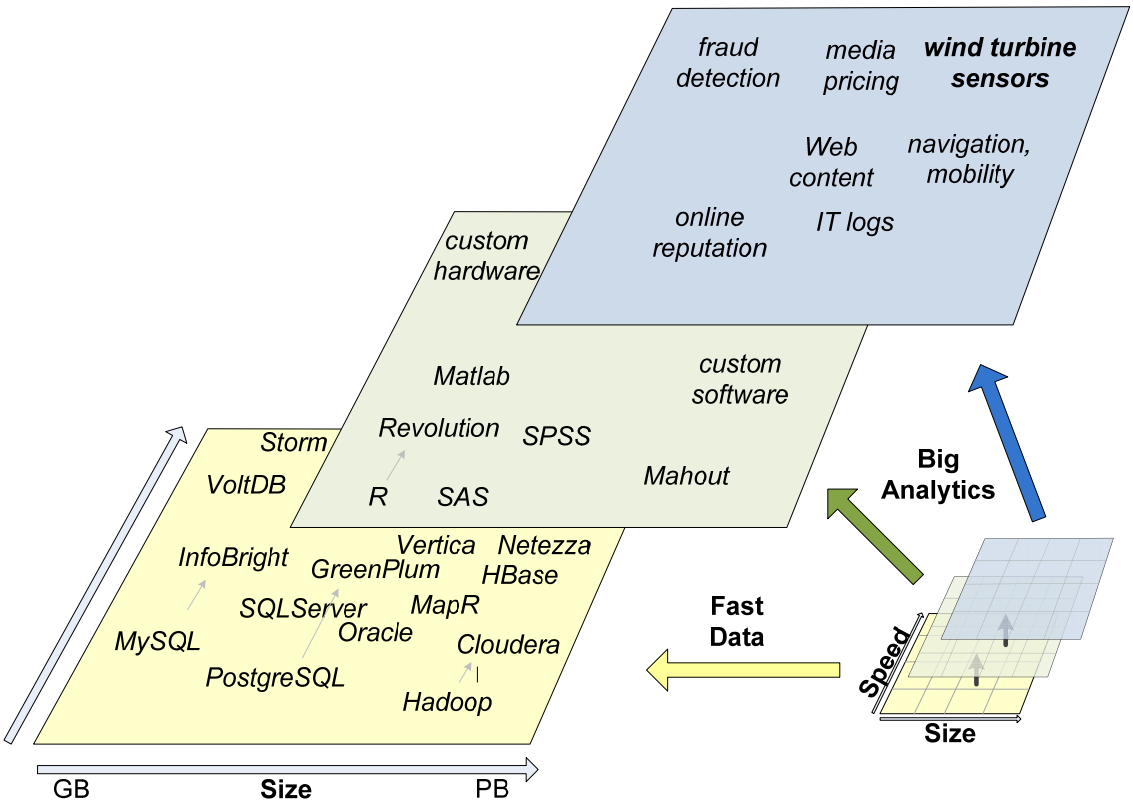

Figure 1: Big data planes with solution provider examples

Another way of categorizing NoSQL tools is based on the data model they support (instead of SQL on relations). The most simple data model handles key-value pairs with simple key-based operators, enabling very fast data access. Project Voldemort and Riak are commonly used key-value stores. There are lots of document databases, supporting large unstructured documents with fast indexing, for example Cassandra. Or, there are tools dedicated to graph data: they support processing of social networks or mobility data; examples include OrientDB or Neo4j. The most similar data model to relational is the so called "wide column" model, where for a given key multiple series of columns are given, without a fixed schema. These were originally used for web user sessions, where the key is a user id and rows contain user interactions. Examples include Google BigTable or Apache HBase.

Performance of NoSQL tools comes at the cost of versatility and consistency. Traditional databases support and require the "ACID" criteria: transaction handling with concurrent but isolated (I), atomic (A), consistent (C) and durable (D) operations. With NoSQL, we satisfy only "BASE": data is basically available (most of the time), states are soft (answers are not consistent all the time), operations are eventually consistent (we reach consistency at some stage). As these new criteria are much weaker than ACID, they are not a perfect match for flight control or banking applications. However, in such scenarios, where erroneous values

appear, concurrent updates are not common, and eventual consistency is sufficient, scalability of NoSQL tools could provide us additional potentials. Wind data analytics is such a typical field: we could gain much more by the increased data sizes behind our analytics than we lose by the reduced consistency criteria.

In the wind data analytics scenario described in the paper Apache Hadoop [21] with Apache Hive [22] were selected as BD solutions. These are actively developed, mature distributed data processing and storage tools used for a lot of analytical and data science BD problems. Hive provides data warehousing environment with SQL interface on top of Hadoop; this enables rapid extension of existing data warehouses.

4. "BIG DATA" SITUATIONS AT WIND TURBINES

Big Data situations at wind turbines can be identified from the following two viewpoints:

- The volume of the data puts too high load on both the local industrial computers and the data centres.
- End user applications must balance between flexibility of querying and quick query response times as distributed systems pose limitations to certain elements of SQL including the join operation.

To illustrate data volumes, an average turbine contains 20-30 sensors, resulting in 60-100 different SCADA signals. With a sampling rate of 1 second with 8 byte values, 1.8 GB raw data per turbine per month are produced, resulting in a big data problem for a typical wind farm having 10-100 turbines, moreover for zones or geographical regions incorporating 5-50 wind farms. It is a *necessity* to apply big data solutions not only to extract valuable knowledge, but even to enable storage of raw data. The task becomes even more challenging, if PLCs collect data with the sampling frequency of some tens of milliseconds.

Big data solutions open up new perspectives, including the following:

- No data has to be dropped in the data chain. For example, both SCADA and PLC data with resolution of some seconds and high frequency condition monitoring signals can be stored centrally.
    - Reports, statistics would become more accurate.
    - In certain situations (failures, stops, unwinding etc.), the reasons could be explored and identified.
    - Turbine technical and operational conditions could be reviewed in detail.
- Using streaming data processing tools, real-time and global calculations could be performed. This enables more accurate, more sophisticated wind farm level alarms and warnings, seeing the whole picture.
- Scalability features of BD solutions eliminate the need for data archiving, all historical data could be taken into account.
- Because extending distributed BD systems is relatively easy and less costly than traditional databases, they can keep up with growing wind farms in a predictable way.

These examples prove that DB aspects are highly relevant in the wind energy field, too.

5. ARCHITECTURES FOR ADVANCED WIND DATA ANALYTICS

Business intelligence software tools are used to enable sophisticated and effective reporting where the back-end of the reporting system is typically a data warehouse [8,9]. Figure 2 introduces architecture alternatives of combined "Big Data" and traditional SQL databases for BD wind data business intelligence.

During operation, wind farms produce input data according to the methods described in Section 2. Both SQL and BD tools can be used to prepare data for reporting.

The first alternative applies a SQL database for aggregating data and to present basic reports for analysis.

Analytical capabilities can be greatly enhanced by dedicated OLAP reporting data marts. These elements are introduced in the second version, which is a classic relational data warehouse with an OLAP reporting tier.

A BD layer appears first in the third alternative architecture, having two roles:

- Extraction, Transformation and Load (ETL): Data coming from wind turbines and wind farms

are collected by the BD system and they are transmitted into the same central database as before.

- Data Warehouse: The BD solution is replacing the data centre providing ETL functionalities and acting as data source for different data marts at the same time.

The last alternative omits SQL tier completely. BD tools with SQL interfaces substitute relational databases, providing the same aggregations and derived data to the data marts. Also, real time alerting functions are implemented by streaming BD tools.

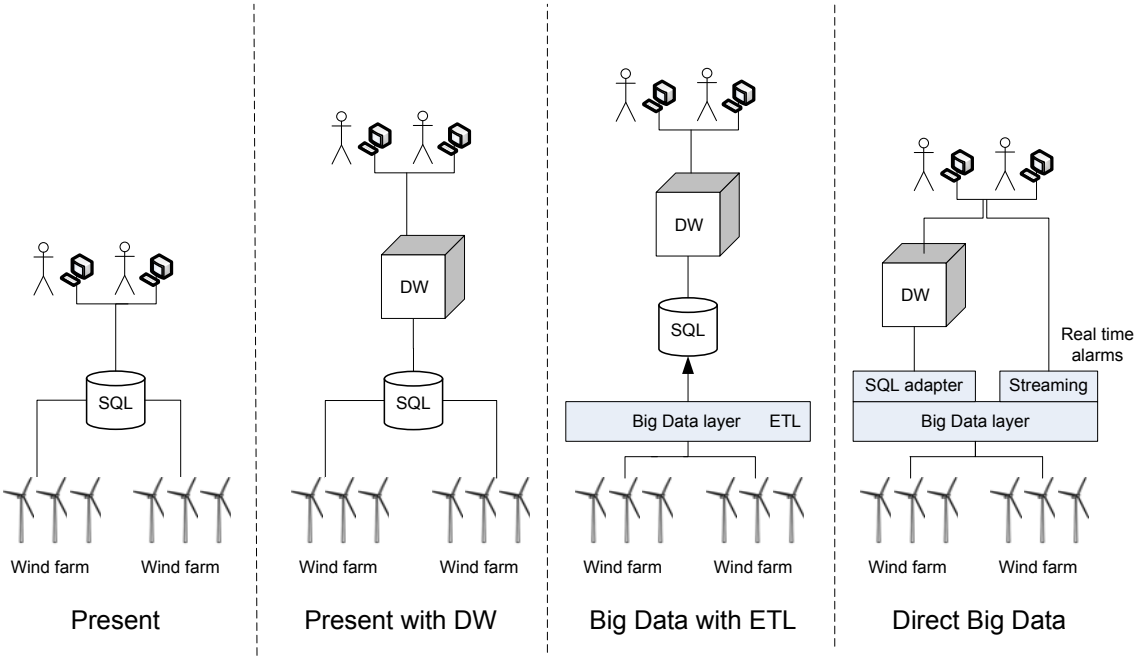

Figure 2: Data processing alternatives for wind farm data

The paper summarizes the experiences in relation of the second and the third cases in the Figure 2 ("Present with DW" and "Big Data with ETL"). The ETL times for loading the same data cubes in these two cases were compared using traditional SQL based database systems (as "Present with DW") and a novel Big Data (NoSQL) solution (as "Big Data with ETL").

6. BIG DATA WIND TURBINE ANALYTICS PROTOTYPE

A prototype BD wind data analytics system was implemented with alternative data flows to simulate and study BD architectures. With this experimental framework scalability properties and limits of the alternative architectures can be evaluated. Data productions of wind farms were simulated in large quantity to answer our main questions:

- "What would happen, if we would have twice, or even 100 times as much turbines, or, much more detailed operation data?"
- "Are BD tools mature enough to substitute IT components in a wind data analytical scenario?"
- Are traditional data warehousing reporting tools applicable on top of new BD solutions?

The data flow, depicted in Figure 3, is as follows. MSSQL database tables substitute real data sources of wind farms, and form the base for both SQL and BD computations. Both two paths aggregate the most detailed data into smaller data cubes. Reporting relies on OLAP analytical user interfaces.

6.1 Benchmarking input dataset

Our prototypes are based on the ten minute average SCADA data that can be extended with more frequent values and also with condition monitoring system measurements.

We simulated

Five wind farms were simulated, resulting in test data resembling the schema of industrial SCADA systems. This data set covers high volumes of commands, alarms and warnings, signals, state and event data for a period of five years. This initial benchmark dataset was generated in a relational database (MS SQL Server), and contains approximately 400 million records.

The initial data set was multiplied by duplicating wind farms to simulate BD situations. Measurements were executed using an appropriate number of wind farms, from one (approx. 80 million records) up to 80 wind farms (approx. 32 billion records).

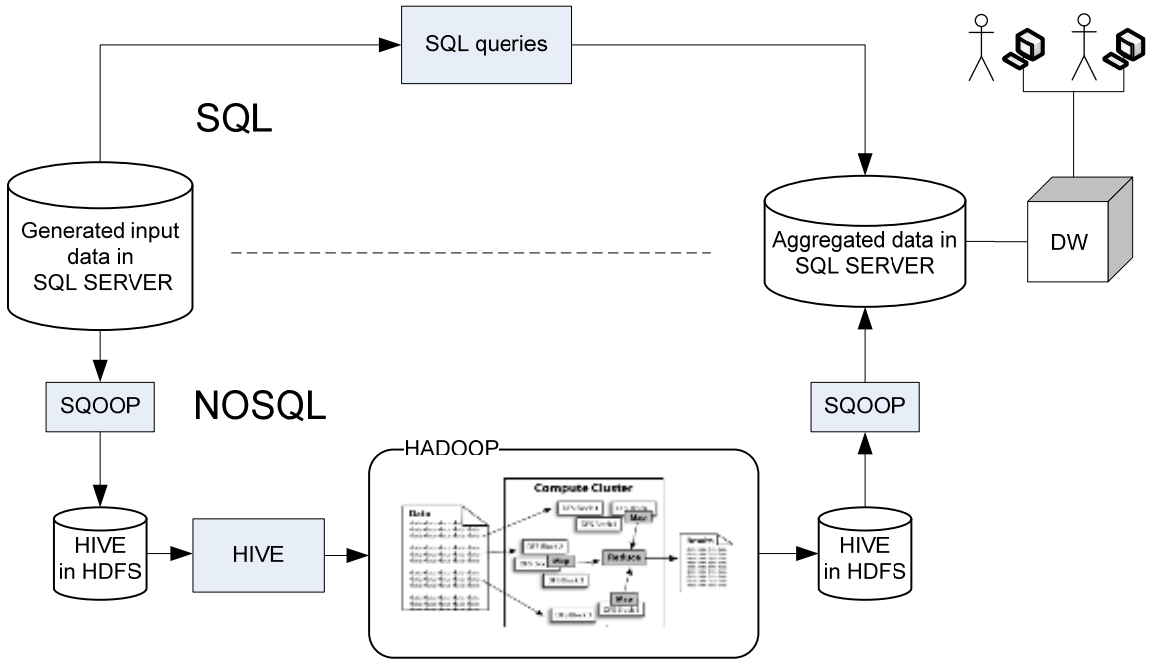

Figure 3: Data flow of the SQL and Big Data solutions, realized by MS SQL Server and a combination of Hadoop and Hive

6.2 Benchmarking operation

A typical data warehousing aggregation operation was selected to test BD and SQL architecture versions. The task is to load a "heavy" aggregate wind farm data cube, combining all types of input data. The aggregation uses all detailed records of the tables in the benchmark dataset, computing for all months and wind farms:

- number of commands,
- number of different alams, warnings and events,
- average, minimum, maximum and standard deviation of the length of all alarms, warnings and events,
- number and average length of different statuses.
- minimum, maximum, average and standard deviation of 8 selected typical, mostly relevant signals.

These are grouping and aggregating procedures, some special mathematical calculations and also using simple join operations.

6.3 Relational database engine, MS SQL Server 2012 (SQL solution)

A straightforward and an advanced ETL process for benchmarking were developed (Figure 3). The straightforward method does the computations in the form of SQL queries using the usual table join, group by constructs and aggregation functions with optimization left to the database engine.

In the advanced method indexes were used, eliminated critical table joins and groupings by prefetching join data and splitting the queries into smaller chunks aligned with group boundaries (techniques common in the relational world to deal with large datasets). By optimization a 2-times speedup was measured. However, in a real life scenario data is coming from the sensors or from the PLC unit on-the-fly in a streaming way. In this setup there is simply no time and capacity to build indexes efficiently. Without indexes the advanced techniques perform even worse than the straightforward method.

Table partitioning is another option that arises in the relational world when dealing with large data volumes. It can improve performance by skipping partitions not needed to satisfy the query. At wind turbine data collection case all the data is needed to do aggregations so all the partitions are needed and not a single one can skipped, consequently, table partitioning was ignored.

6.4 Hadoop-Hive based test environment (Big Data solution)

The Big Data layer (Figure 3) is composed from the industry-wide accepted Apache Hadoop framework and the Apache Hive data warehouse system.

The Hadoop framework allows for the distributed processing of large data sets using the so-called map-reduce programming model. The framework is scalable from single server up to thousands of nodes. All nodes have local storage and computation. The system is designed to provide error-free operation. This is achieved using software error handling mechanism at the application layer instead a high-availability hardware layer. The framework is capable of handling dynamic addition, removal and failure of nodes with data replication. Two main component of the system are the map-reduce job scheduling framework and the distributed file system (HDFS).

On top of these frameworks operates the Apache Hive data warehouse system that act as a client to the Hadoop. It provides a convenient SQL-like language (HiveQL) to perform data aggregation, ad-hoc querying and analysis of large datasets. The tables are stored in the HDFS and the SQL queries are mapped to map-reduce operations executed by the Hadoop framework. The functionalities of the HiveQL can be extended by custom functions thus performing non-standard calculations.

The data export-import between Hive and MSSQL is handled by the Apache Sqoop tool.

6.5 Experiments

Three different environments for our scalability tests was set up, one to test SQL, and two to test NoSQL solutions.

A mid-range server, having 4 Intel Xeon processor cores, 10 GB memory and older disks with RAID was running a MSSQL Server 2008. This setup was used to test traditional relational data warehousing methods. Also the common reporting OLAP frontend was running here. Since the MSSQL server used limited resources a Linux cluster was created with only two mid-range nodes, each having 8 Intel Xeon cores and 12TB of HDFS storage.

One of the main points of Hadoop is the easy extendibility; we validated this by moving our Hadoop setup to a 48 node Linux cluster. Here each node contained 2 Intel Pentium processor cores, 4GB memory and older disks without RAID.

Figure 4 shows the running times of generating the aggregated data cube on these three architectures. As hardware and software environments were highly different, times are not comparable directly. It was examined were the shapes of the series vary: these indicate how the tools scale with data volumes.

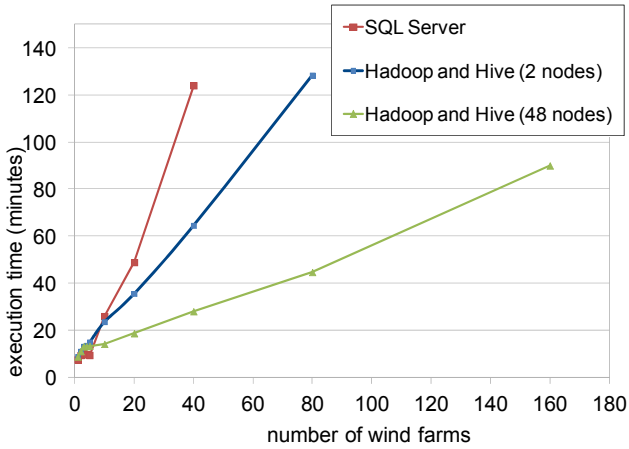

Figure 4: Scalability of different cube aggregator solutions

The SQL solution ran almost in linear time up to 40 farms (approximately 3.2 billion records). Farther up processing times drastically fall. The reason is that we reached the hardware, especially the memory limits of the given database.

Both NoSQL series scale linearly with the input size (number of simulated wind farms) examined, while computing times of the larger cluster are significantly smaller. No sudden relapse was experienced. Therefore, for a given input size the infrastructural requirements are predictable. Furthermore, the cost of extending a Hadoop cluster comes from adding new identical nodes. For relational databases, extending resources might be more costly. Adding memory or CPU can postpone the point where performance degrades, but above that point, setting up powerful distributed relational database appliances is a costly process.

For smaller data sets MSSQL outperforms Hive solutions. The reason for that is that the Hadoop framework has a constant job initialization penalty.

7. WIND TURBINE BUSINESS INTELLIGENCE SYSTEM

SCADA information can be analysed by various experts and managers based on specific reports. Reporting is based on the individual cubes of the data mart(s). These are designed according to common OLAP conceptual data modelling techniques [8, 9]. The data cubes used in our prototype system are:

- Commands
- Alarms & Warnings & Events
- Statuses
- Signals
- SCADA (this cube “integrates” information from the previous cubes)
- SCADA slice (this cube is the same as the previous SCADA cube but it contains only a small, selected fraction of its data content)
- Aggregate wind farm information (this cube contains the same information as the SCADA cube but in a significantly selected and aggregated form – for information integration and performance reasons as described before)

All cubes have its own set of dimensions with a given granularity. The dimensions used for the data cubes are the following:

- Wind farm
- Wind turbine
- Status
- Category (incorporating codes for measured and calculated signals, alarms and warnings, etc.)
- Month
- Date
- Time
- Source (this is a technical dimension not a user application specific one)

The On-line Analytical Processing (OLAP) part of the architectures (see Figure 2) is a data mart realized in Microsoft SQL Server 2012 Analysis Services (MSSAS). Our prototype reporting interface is Ms Excel with direct connection to MSSAS cubes. Of course, many other (e.g. web based) reporting solutions can be used for report building ensuring with the same communication protocol.

Figure 4-7 show sample reports of the system. (The diagrams serve only as illustrative examples they are based on generated and not on real SCADA dataset).

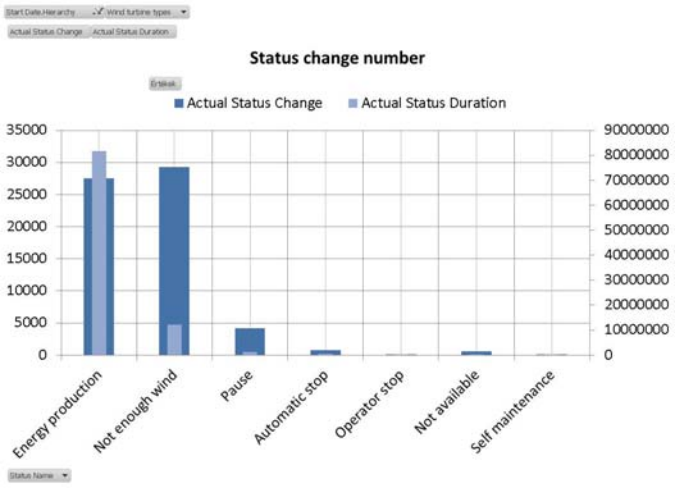

Figure 4: Report based on the Status cube representing the distribution in the number of statuses and the related duration being in the given status

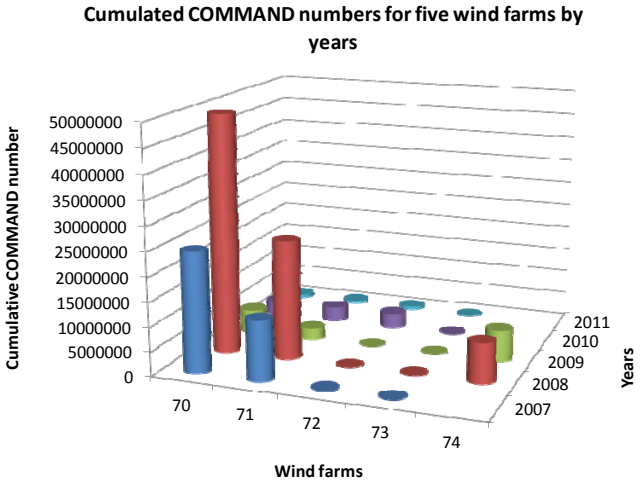

Figure 5: Report based on the Commands cube representing the distribution in the cumulative number of commands of five different turbines over five years

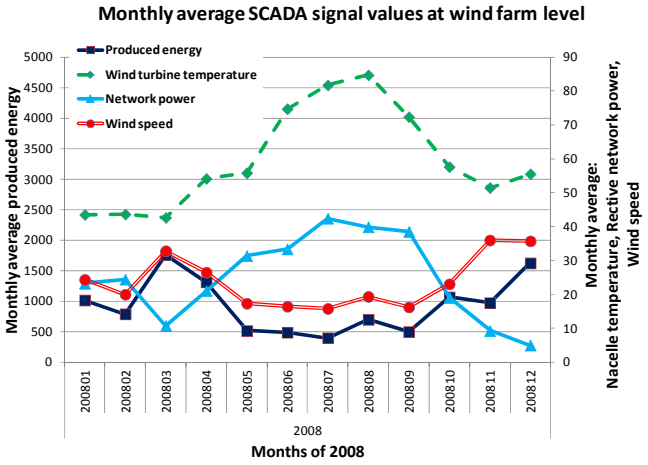

Figure 6: Report based on the Signals cube representing the monthly averages of four different SCADA signals

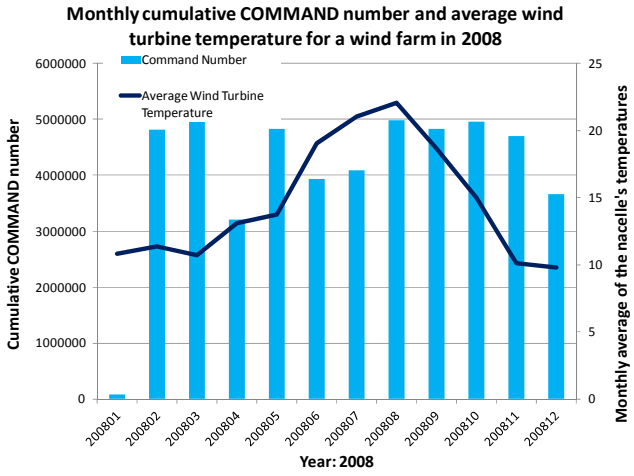

Figure 7: The report is based on the Aggregated SCADA cube and represents two kinds of information in the same time: command number and average wind turbine temperature

Each of these diagrams has the flexibility of the basis BI system, for example drill-down, dimension change, aggregation and selection. Consequently, a great variety of management and expert reports on the behaviour and working conditions of wind turbines can be prepared easily and quickly.

8. CONCLUSIONS

In this paper it was introduced how Big Data tools provide value for wind farm operation and maintenance. After briefly introducing the problem of collecting and analysing wind turbine data, Big Data architecture alternatives were described. We validated the idea of substituting relational database engines with NoSQL tools for data intensive tasks using an experimental prototype analytical system. *The measurements showed that Hadoop and Hive as applied Big Data solutions could provide a cost effective alternative of traditional data warehousing and ETL tools processing SCADA data of many wind farms*. An OLAP analytical frontend for wind data is also presented, which can be built on both SQL and NoSQL solutions.

As a future work we plan to extend our experiments towards streaming Big Data tools for real-time analytics. Experiments will also be extended with other data-heavy tasks besides the construction of the selected data cube presented.

ACKNOWLEDGEMENTS

The research is supported by the “Big Data” business intelligence reference group project of the Computer and Automation Research Institute of the Hungarian Academy of Sciences [7] and OTKA NK 105645.

REFERENCES

1. The Emerging Big Data, Intelligent Information Management DG INFSO/E2 Objective, Info day, ICT-2011.4.4
2. Frankel, F., Reid, R., “Big data: Distilling meaning from data”, *Nature*, 2008; 455, 28-30
3. Blanco, MI., “The economics of wind energy”, *Renewable and Sustainable Energy Reviews* 2009; 13, 1372-1382
4. Faulstich, S., Kühn, P. and Lyding, P., “Establishing a common Database for Maintenance Optimisation”, *SKF Wind Farm Management Conference 2011'How to reduce costs and improve revenue and profitability*', Barcelona, Spain, 11.-12.05.2011.
5. Guo, H., Watson, S., Tavner, P. and Xiang, J. “Reliability analysis for wind turbines with incomplete failure data collected from after the date of initial installation”, *Reliability Engineering and System Safety* 2009; 94,1057-1063
6. Hameed, Z., Hing, YS., Cho, YM., Ahn, SH., Song, CK., ”Condition monitoring and fault detection of wind turbines and related algorithms: A review”, *Renewable and Sustainable Energy Reviews* 2009; 13, 1-39
7. http://bigdatabi.sztaki.hu/
8. W. H. Inmom, *Building the Data Warehouse, 4nd Edition*, Book, John Wiley & Sons, Inc., ISBN-13: 978-0764599446, 2007
9. R. Kimball, M. Ross, W. Thornthwaite, J. Mundy and B. Becker, *The Data Warehouse Lifecycle Toolkit, 2nd Edition*, Book, John Wiley & Sons, Inc., ISBN-13: 978-0470149775, 2008
10. Korbicz, J., Koscielny. J.M., Kowalczuk Z., Cholewa, W. and Karbicz J., “Fault Diagnosis: Models, Artificial Intelligence, Applications”. *SpringerVerlag, ISBN:3540407677*, 2004
11. Kusiak, A., Zheng, H., Song, Z., “Models for monitoring wind farm power”, *Renewable Energy,* 2009, 34, 583-590
12. J. Markard and R. Petersen, *The offshore trend: Structural changes in the wind power sector*, Energy Policy 2009, 37, 3545-3556
13. Paliwal, M. and Kumar UA., “Neural networks and statistical techniques: A review of applications”, *Expert Systems with Applications,* 2009, 36., 2-17
14. Project report: CONMOW: Condition Monitoring for Offshore Wind Farms, State of the art condition monitoring techniques suitable for wind turbines and wind farms, 2005
15. Sainz E., Llombart A. and Guerrero JJ. “Robust filtering for the characterization of wind turbines: Improving its operation and maintenance”, *Energy Conversion and Management,* 2009, 50., 2136-2147
16. Viharos, Zs. J. and Kemény, Zs., “AI techniques in modelling, assignment, problem solving and optimisation”, *AI-METH 2005 - Artificial Intelligence Methods*, Gliwice, Poland, November 16-18, 2005., 225-230
17. Viharos, Zs. J., Monostori, L., Novák, K., Tóth, G., Csongrádi, Z., Kenderesy, T., Sólymosi, T., Lőrincz, Á., Koródi, T., “Monitoring of complex production systems, in view of digital factories”, *Proceedings of the XVII IMEKO World Congress - Metrology in the 3rd Millennium*, Dubrovnik, Croatia, June 22-27, 2003, 1463-1468
18. Wachla, D., Moczulski, W.A. “Identification of dynamic diagnostic models with the use of methodology of knowledge discovery in databases”, *Engineering Applications of Artificial Intelligence 2007*, 20., 699-707
19. Billions and billions: big data becomes a big deal, Deloitte: http://www.deloitte.com/view/en_GX/global/industries/technology-media-telecommunications/tmt-predictions-2012/technology/70763e14447a4310VgnVCM100000 01a56f00aRCRD.htm
20. Twitter Storm. http://engineering.twitter.com/2011/08/stormis-coming-more-details-and-plans.html.
21. T. White. Hadoop: The Definitive Guide. Yahoo Press, 2010.
22. Apache Hive. http://hive.apache.org/
23. Gilbert, Seth and Lynch, Nancy, “Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services”. *SIGACT News*, Volume 33., 2002
24. Eric Redmond, Jim R. Wilson, “Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement”. *O'Reilly, 2012, ISBN :1934356921,* 2012