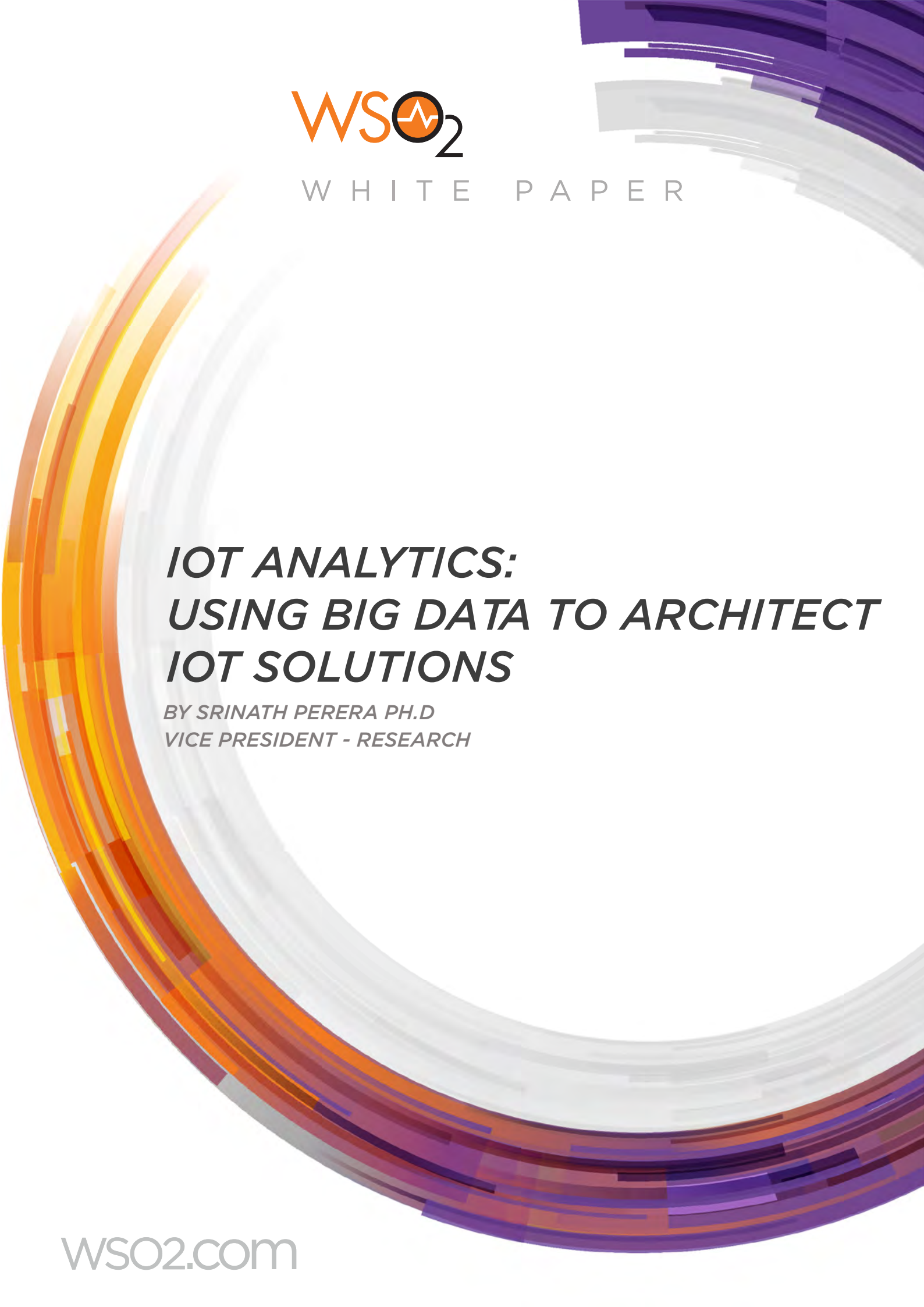

WHITE PAPER

# IOT ANALYTICS: USING BIG DATA TO ARCHITECT IOT SOLUTIONS

BY SRINATH PERERA PH.D
VICE PRESIDENT - RESEARCH

wso2.com

# TABLE OF CONTENTS

# 1. INTRODUCTION

A typical IoT system would comprise the architecture depicted in Figure 1; sensors would collect data and transfer them to a gateway, which in turn would send them to a processing system (analytics cloud). The gateway can choose either to or not summarize or preprocess the data. The connection between sensors and gateway would be via Radio Frequency (e.g. Zigbee), BLE, Wifi, or even wired connections. Often, the gateway is a mobile phone.

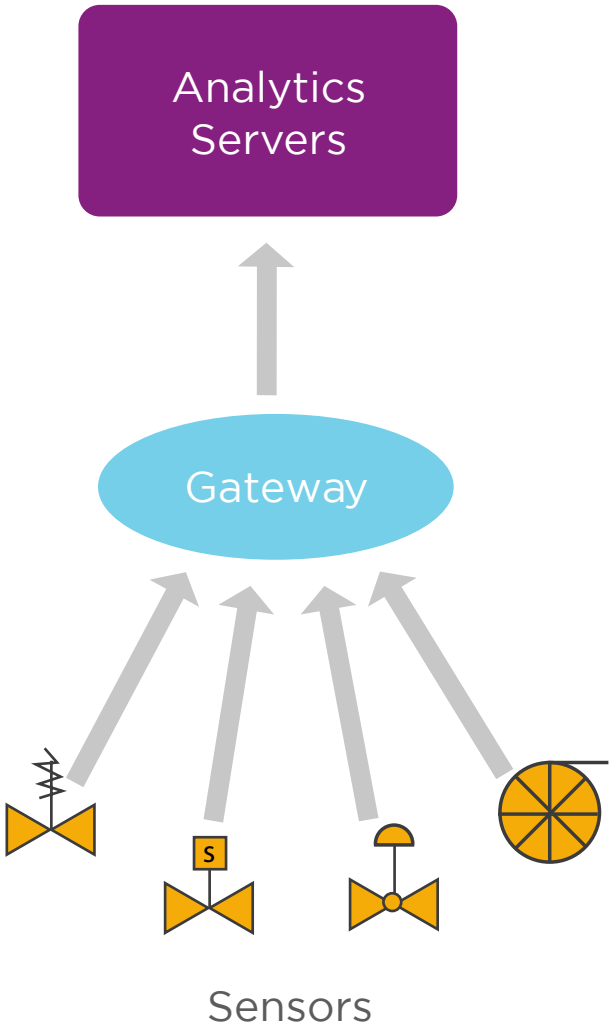


*Figure 1*

The connection from the gateway to the analytic servers would be via the Internet, LAN, or WiFi connection, and it will use a higher level protocol, such as MQTT or CoAp[1]. Given that the focus of this paper is on IoT analytics, we won't be delving too much into devices and connectivity. Assuming that part is done, how hard is it to figure out IoT analytics? Is it just a matter of offloading the data into one of the IoT analytics platforms or are there hidden surprises?

1 Refer to IoT Protocols

This white paper aims to explain the challenges and discuss how big data analytics is used to architect IoT solutions. 'Big data' efforts have solved many IoT analytics challenges, particularly system challenges related to large-scale data management, learning, and data visualizations. Data for 'big data,' however, came mostly from computer-based systems (e.g. transaction logs, system logs, social networks, and mobile phones). IoT data, in contrast, will come from the natural world, would be more detailed, fuzzy, and large. The nature of that data, assumptions, and use cases differ between old big data and new IoT data. IoT analytics designers can build on top of big data, yet the work would be far from being done. Let's analyze a few key requirements that you would need to think about first.

## 2. IOT ANALYTICS - KEY REQUIREMENTS

### *HOW FAST DO YOU NEED RESULTS?*

This would generally depend on how fast you need results from the data gathered and your design changes and would vary according to each use case. You would need to consider if the value of your insights (i.e. results) would degrade over time and how fast this would happen, e.g. if you're going to improve the design of a product using data, then you could wait days or even weeks. On the other hand, if you're dealing with stock markets and other similar use cases where winner takes all, the milliseconds are a big deal.

Speed comes in several levels

- A few hours – send your data into a data lake and use a MapReduce technology, such as Hadoop or Spark for processing
- A few seconds – send data into a stream processing system (e.g. Apache Storm or Apache Samza), an in-memory computing system (e.g. VoltDB, Sap Hana), or an interactive query system (e.g. Apache Drill) for processing
- A few milliseconds – send data to a system like complex event processing where records are processed one by one and produce very fast outputs.

Figure 2 summarizes those observations. It's also likely that some use cases will fall under more than one in which case you would need to use multiple technologies.

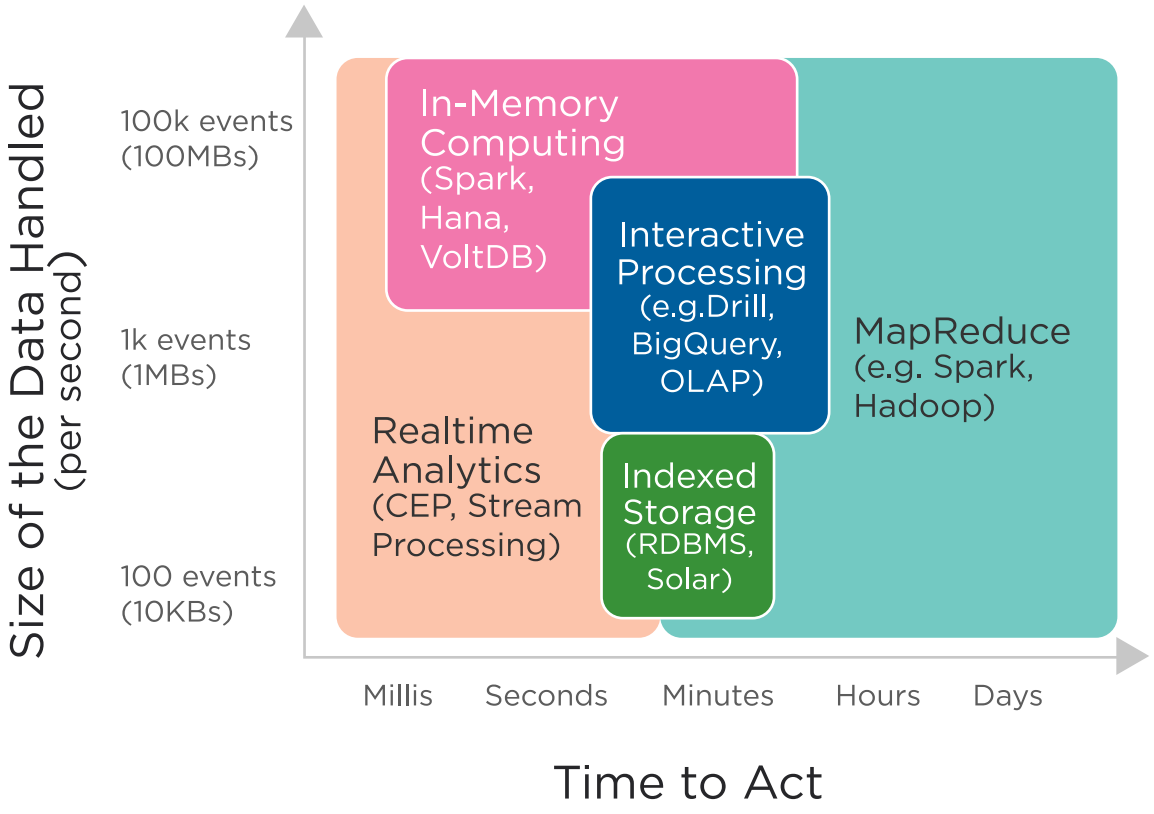


*Figure 2*

### *HOW MUCH DATA SHOULD YOU KEEP?*

Next, we should decide how much data to keep and in what form. It is a tradeoff between cost versus potential value of data and associated risks. Data is valuable. In recent times, companies have been acquired just for their data, while Google and Facebook have gone to extraordinary lengths to access data. Moreover, you may find a bug or improvement in the current algorithm, and might want to go back and rerun the algorithm on old data. Yet, all decisions must be made considering the bigger picture and current limits. The choices are as follows:

- Keep all the data and save it to a data lake (the argument is that disk is cheap)
- Process all the data in a streaming fashion and not keep any data at all
- Keep a processed or summarized version of the data; however, it is possible that you cannot recover all the information from the summaries later

The next question is where to do the processing and how much of that logic you should push towards the sensors. Pushing logic towards sensors will let your system scale further. There are three options as follows:

- Do all processing at analytics servers
- Push some queries into the gateway
- Push some queries down to sensors as well

The IoT community already has the technology to push the logic to gateways. Most gateways are fully-fledged computers or mobile phones, and they can run higher level logic, such as SQL-like CEP queries. However, if you want to push code into sensors, in most cases, you would have to write custom logic using a lower level language like Arduino C. Another associated challenge is deploying, updating, and managing queries over time. If you choose to put custom low-level filtering code into sensors, there's a likelihood this may lead to deployment complexities in the long run.

## 3. ANALYTICS: HINDSIGHT, INSIGHT, OR FORESIGHT?

Hindsight, insight, and foresight are three questions that come to mind when dealing with data; to know what happened, to understand what happened, and to predict what will happen.

Hindsight is possible with aggregations and applied statistics. You can aggregate data by different groups and compare those results using statistical techniques, such as confidence intervals and statistical tests. A key component is data visualization that will show related data in context[2].

Insight and foresight would require machine learning and data mining. This includes finding patterns, modeling current behavior, predicting future outcomes, and detecting anomalies. Refer to data science and machine learning tools (e.g. R, Apache Spark MLLib, WSO2 Machine Learner, GraphLab) for a deeper understanding.

IoT analytics will pose new types of problems and demand more focus on some existing problems. Some problems that are likely to play a key role in IoT analytics are as follows:

### *TIME SERIES PROCESSING*

Most IoT data are collected via sensors over time. Hence, they are time series data, and often most readings are autocorrelated, e.g. a temperature reading is often highly affected by the earlier time step's reading. However, most machine learning algorithms (e.g. Random Forests or SVM) do not consider autocorrelation. Hence, those algorithms would often do poorly while predicting using IoT data.

This problem has been extensively studied under time series analysis (e.g. ARIMA model). Moreover, in recent years, Recurrent Neural Networks (RNN) has shown promising results with time series data. However, widely used big data frameworks, such as Apache Spark and Hadoop, do not support these models yet. The IoT analytics community has to improve these models, build new models when needed, and incorporate them into big data analytics frameworks[3].

---

2 Refer to Napoleon's March and Hans Rosling's famous Ted talk

3 For more information on the topic, refer to the article Recurrent neural networks, Time series data and IoT: Part I.

### *SPATIOTEMPORAL ANALYSIS AND FORECASTS*

Similarly, most IoT data would include location data, making them spatiotemporal data sets (e.g. geospatial data collected over time). Just like time series data, these models would be affected by the spatial neighborhood. You would need to explore and learn spatiotemporal forecasting and other techniques and build tools that support them. Among related techniques are GIS databases (e.g. Geotrelis), and panel data analysis. Moreover, machine learning techniques, such as Recurrent Neural networks might also be used[4].

### *ANOMALY DETECTIONS*

Many IoT use cases like predictive maintenance, health warnings, finding plug points that consumes too much power, optimizations, etc., depend on detecting anomalies. Anomaly detection poses several challenges.

- Lack of training data – most use cases would not have training data, and hence unsupervised techniques, such as clustering, should be used
- Class imbalance – Even when training data is available, often there will be a few dozen anomalies that exist among millions of regular data points. This problem is generally handled by building an ensemble of models where each model is trained with anomalous observations and resampled data from regular observations.
- Click and explore – after detecting anomalies, they must be understood in context and vetted by humans. Tools, therefore, are required to show those anomalies in context and enable operators to explore data further starting from the anomalies. For example, if an anomaly in a turbine is detected, it is useful to see that anomaly within regular data before and after the anomaly as well as to be able to study previous similar cases.

## *4. ACTING ON IOT DATA*

Once the data has been analyzed and actionable insights have been identified, you would need to decide on the next course of action. There are several choices to this end.

4 Refer to Application of a Dynamic Recurrent Neural Network in Spatio-Temporal Forecasting

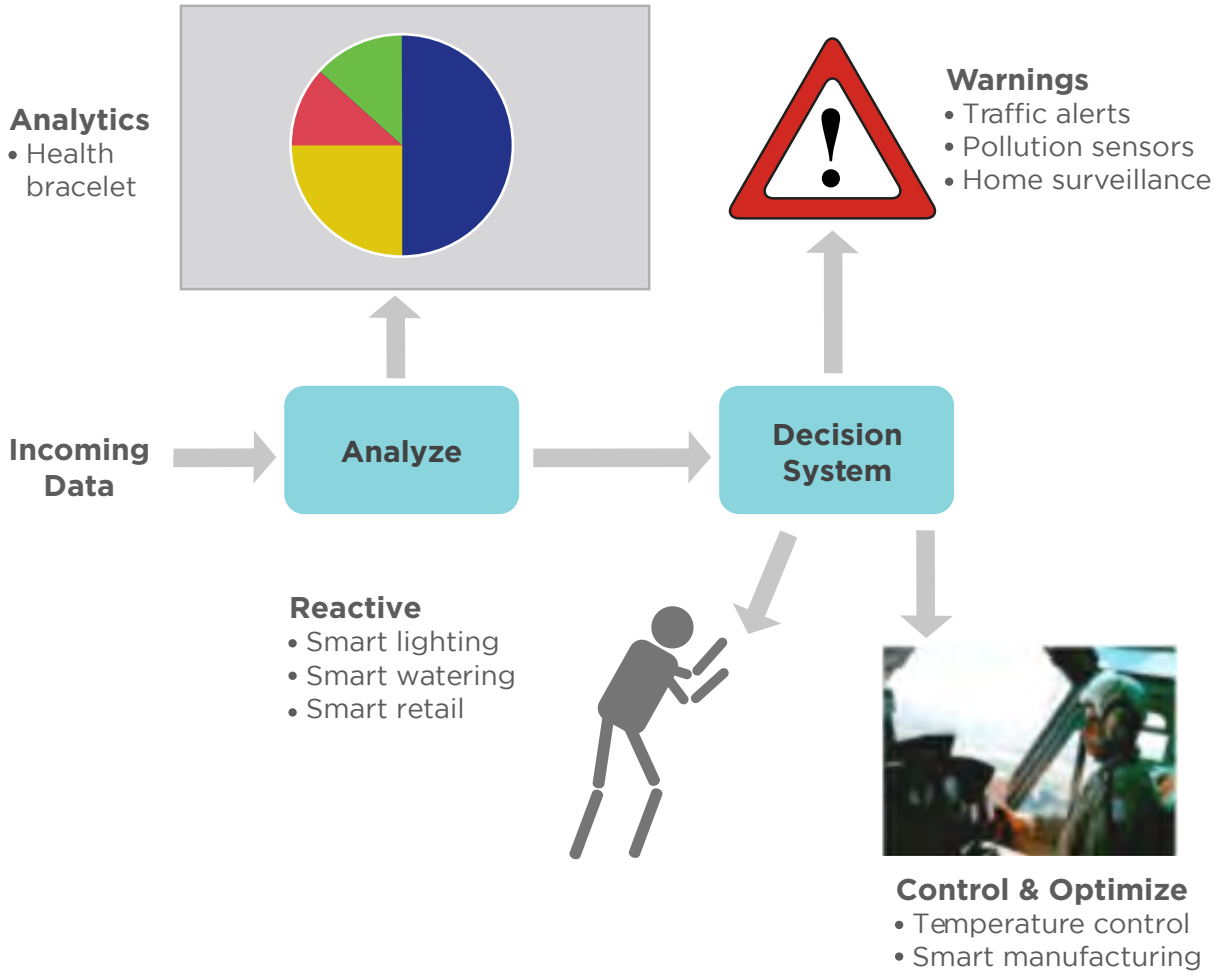


*Figure 3*

- Visualize the results – build a dashboard that shows the data in context and let users explore, drill-down, and carry out root cause analysis
- Alerts – detect problems and notify the user via email, SMS, or pager devices. Your primary challenge would be false positives that would severely affect the operator's trust on the system. Finding the balance between false positives and ignoring actual problems will be tricky
- Carrying out actions – the next level is independent actions with open control loops; however, unlike in the former case, the risk of a wrong diagnosis could have catastrophic consequences. Until we have a deeper understanding of the context, use cases would be limited to simple applications like turning off a light, adjusting heating, etc. where the associated risk is relatively minor
- Process and environment control – this is the holy-grail of automated control. The system would continuously monitor and control the environment or the underlying process in a closed control loop. The system has to understand the context, the environment, and should be able to work around failures of actions. Much related work has been carried out under theme autonomic computing 2001-2005 although only a few use cases were eventually deployed. Real-life production deployment of this class, however, are several years away due to associated risks. NEST and Google Auto driving Car are possibly first examples of such systems.

In general, the move towards automation is prompted by the need for fast responses (e.g. algorithmic trading). More automation can be cheaper in the long run, but is likely to be complex and expensive in the short run. As evidenced by stock market crashes, the associated risks cannot be underestimated. It is worth noting that carrying out automation with IoT will be harder than big data automation use cases. Most big data automation use cases either monitor computer systems or controlled environments like factories. In contrast, IoT data would often be fuzzy and uncertain. It is one thing to monitor and change a variable in automatic price setting algorithm. However, automating a use case in the natural world (e.g. an airport operation) is different altogether. If you decide to pursue the automation route, you need to spend a significant amount of time to understand, test, and re-test the scenarios.

## 5. UNDERSTANDING IOT USE CASES

Now that you've possibly got a good understanding of how IoT analytics work, let's discuss the shape of common IoT data sets and use cases that arise from these.

Data from most devices would have the following fields:

- Timestamp
- Location, grouping, or proximity data
- Several readings associated with the device, e.g. temperature, voltage and power, rpm, acceleration, and torque, etc.

The first use case is to monitor, visualize, and alert about a single device data. This use case focuses on individual device owners. However, more interesting use cases occur when you look at devices as part of a larger system like a fleet of vehicles, buildings in a city, a farm, etc. Among the aforementioned fields, time and location will play a key role in most IoT use cases. By using these two, you can categorize most use cases into two classes: stationary dots and moving dots.

### *STATIONARY DOTS*

Among examples of 'stationary dot' use cases are equipment deployments (e.g. buildings, smart meters, turbines, pumps, etc). Their location is useful only as a grouping mechanism, but the main goal is to monitor an already deployed system in operation.
Some of the use cases are as follows:

- View of the current status, alerts on problems, drill down, and root cause analysis
- Optimization of current operations
- Preventive maintenance
- Surveillance

### *MOVING DOTS*

Among examples of moving dot use cases are fleet management, logistic networks, wildlife monitoring, monitoring customer interactions in a shop, traffic, etc. The goal of these use cases is to understand and control movements, interactions, and behavior of participants as illustrated in this screencast - [Screencast] Analyzing Transport for London Data with WSO2 CEP.

Some examples are as follows:

- Sports analytics (refer to Real Time Analytics for Football - a sports analytics use case built using data from an actual football game)
- Geofencing and speed limits
- Monitoring customer behavior in a shop, guided interactions, and shop design improvements
- Visualizing (e.g. time-lapse videos) of movement dynamics
- Surveillance
- Route optimizations

For both types of use cases, it's possible to build generic extensible tools that provide an overall view of the devices and provide out-of-the-box support for some of these. However, specific machine learning models, such as anomaly detection, would need expert intervention for best results. Such tools, if done right, could facilitate reuse, reduce cost, and improve the reliability of IoT systems. It is worth noting that this is one of the things that the 'big data' community did right. A key secret of big data success so far has been the availability of high quality, generic open source middleware tools. There's also great potential for companies that focus on specific use cases or classes of use cases, e.g. Scanalytics focuses on foot traffic monitoring and Second spectrum focuses on sport analytics. Although expensive, they would provide an integrated, ready-to-go solution. IoT system designers have a choice to either opt for a specialized vendor or build on top of open source tools (e.g. Eclipse IoT platform, WSO2 Analytics Platform).

## 6. AN OVERVIEW OF THE WSO2 ANALYTICS PLATFORM

If you are looking to build your own analytics platform on top of a open source platform, WSO2 analytics platform can be a great fit.

The WSO2 Analytics platform combines into one integrated platform real-time and batch analysis of data with predictive analytics via machine learning to support the multiple demands of IoT solutions, as well as mobile and web apps. It also has the capability to organize and analyze data that would have been previously inaccessible or unusable. Moreover, it builds on the fast performance of the open source Siddhi CEP engine developed by WSO2 by adding streaming regression and anomaly detection operators to facilitate fraud and error detection.

As part of WSO2's analytics platform, WSO2 Data Analytics Server 3.0 has the ability to analyze both data in motion and data at rest from the same software. The comprehensive platform provides a single solution that enables developers to build systems and applications that collect and analyze information and communicate the results. It has been designed to treat millions of events per second, and is therefore capable to handle the volumes in big data and IoT projects.

## 7. CONCLUSION

As discussed, there are different aspects of an IoT analytics solution, particularly challenges you would need to consider when building or choosing an IoT analytics solutions. Big data has solved many IoT analytics challenges, especially system challenges related to large-scale data management, learning, and data visualizations. Nevertheless, significant thinking and work is required to match IoT use cases to analytics systems. Among the highlights are how fast you need results, i.e. real-time or batch or a combination; deciding how much data should be kept based on use cases and the incoming data rate; deciding between aggregation and learning methods; and identifying your response once an actionable insight has been defined.

## ABOUT THE AUTHOR

**Srinath Perera Ph.D**
Vice President - Research
WSO2

Srinath has been working with large-scale distributed systems and parallel computing for about 10 years and is a co-architect of the WSO2 complex event processing engine. He has also been a long-standing open source contributor; he is a co-founder of Apache Axis2 (open source Web Service engine), a member of the Apache Software foundation, and Committer for Apache Geronimo (J2EE Engine) and Apache Airavata. Srinath has authored two books about MapReduce, many technical articles (e.g. at IBM Developerworks, InfoQ), and peer-reviewed over 20 articles. He holds a B.Sc. from University of Moratuwa, Sri Lanka, and a Ph.D. from Indiana University, Bloomington, USA.

## ABOUT WSO2

WSO2 is the only company that provides a completely integrated enterprise application platform for enabling a business to build and connect APIs, applications, web services, iPaaS, PaaS, software as a service, and legacy connections without having to write code; using big data and mobile; and fostering reuse through a social enterprise store. Only with WSO2 can enterprises use a family of governed secure solutions built on the same code base to extend their ecosystems across the cloud and on mobile devices to employees, customers, and partners in anyway they like. Hundreds of leading enterprise customers across every sector—health, financial, retail, logistics, manufacturing, travel, technology, telecom, and more—in every region of the world rely on WSO2's award-winning, 100% open source platform for their mission-critical applications. To learn more, visit http://wso2.com or check out the WSO2 community on the WSO2 Blog, Twitter, LinkedIn, and Facebook.

Check out more **WSO2 White Papers** and **WSO2 Case Studies.**

For more information about WSO2 products and services, please visit **http://wso2.com** or email **bizdev@wso2.com**

WSO2.com