IoT Data Analytics Challenges

As IoT has grown and evolved, it has become clear that traditional data analytics solutions were not always adequate.

Traditional data analytics typically employs a standard RDBMS and corresponding tools, but the world of IoT is much more demanding.

Relational databases often struggle with the nature of IoT data.

IoT data places two specific challenges on a relational database as follows:

Scaling problems: Due to the large number of smart objects in most IoT networks that continually send data, relational databases need to grow incredibly large and very quickly. This can result in performance issues that can be costly to resolve, often requiring more hardware and architecture changes.

Volatility of data: With relational databases, it is critical that the schema be designed correctly from the beginning. Changing it later can slow or stop the database from operating.

Due to the lack of flexibility, revisions to the schema must be kept at a minimum. IoT data, however, is volatile in the sense that the data model is likely to change and evolve over time.

A dynamic schema is often required so that data model changes can be made daily or even hourly.

To deal with challenges like scaling and data volatility, a different type of database, known as NoSQL, is being used. Structured Query Language (SQL) is the computer language used to communicate with an RDBMS.

In addition to the relational database challenges that IoT imposes, with its high volume of smart object data that frequently changes, IoT also brings challenges with the live streaming nature of its data and with managing data at the network level. Streaming data, which is generated as smart objects transmit data, is challenging because it is usually of a very high volume, and it is valuable only if it is possible to analyze and respond to it in real-time. Real-time analysis of streaming data allows you to detect patterns or anomalies that could indicate a problem or a situation that needs some kind of immediate response. To have a chance of affecting the outcome of this problem, you naturally must be able to filter and analyze the data while it is occurring, as close to the edge as possible. The market for analyzing streaming data in real-time is growing fast.

Major cloud analytics providers, such as Google, Microsoft, and IBM, have streaming analytics offerings, and various other applications can be used in house. (Edge streaming analytics is discussed in depth later in this chapter.) Another challenge that IoT brings to analytics is in the area of network data, which is referred to as network analytics. With the large numbers of smart objects in IoT networks that are

communicating and streaming data, it can be challenging to ensure that these data flows are effectively managed, monitored, and secure. Network analytics tools such as Flexible NetFlow and IPFIX provide the capability to detect irregular patterns or other problems in the flow of IoT data through a network. Network analytics, including both Flexible NetFlow and IPFIX, is covered in more detail.

Data acquiring

Having learnt about devices, devices-network data, messages and packet communication to the Internet, let us understand the functions required for applications, services and business processes at application-support and application layers. These functions are data acquiring, data storage, data transactions, analytics, results visualisations, IoT applications integration, services, processes, intelligence, knowledge discovery and knowledge management.

Data Acquiring

1. Data Generation

Data generates at devices that later on, transfers to the Internet through a gateway. Data generates as follows:

Passive devices data: Data generate at the device or system, following the result of interactions. A passive device does not have its own power source. An external source helps such a device to generate and send data. Examples are an RFID or an ATM debit card. The device may or may not have an associated microcontroller, memory and transceiver. A contactless card is an example of the former and a label or barcode is the example of the latter.

Active devices data: Data generates at the device or system or following the result of interactions. An active device has its own power source. Examples are active RFID, streetlight sensor or wireless sensor node. An active device also has an associated microcontroller, memory and transceiver.

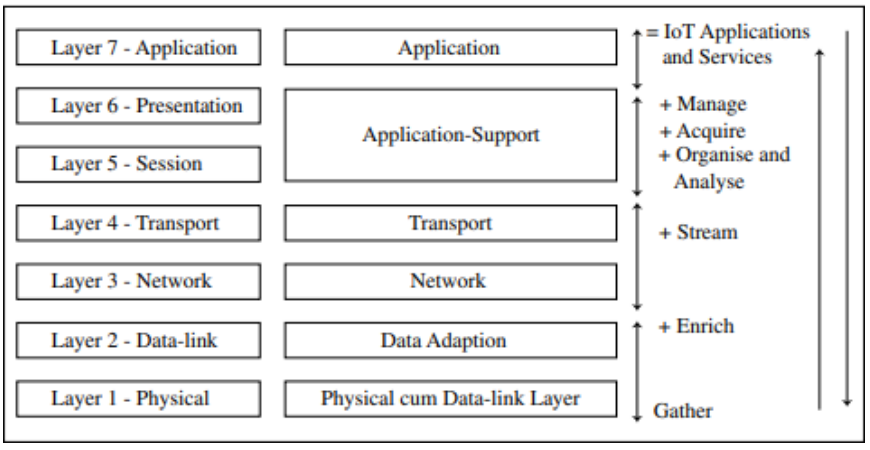
Event data: A device can generate data on an event only once. For example, on detection of the traffic or on dark ambient conditions, which signals the event. The event on darkness communicates a need for lighting up a group of streetlights (Example 1.2). A system consisting of security cameras can generate data on an event of security breach or on detection of an intrusion. A waste container with associate circuit can generate data in the event of getting it filled up 90% or above. The components and devices in an automobile generate data of their performance and functioning. For example, on wearing out of a brake lining, a play in steering wheel and reduced air- conditioning is felt. The data communicates to the Internet. The communication takes place as and when the automobile reaches near a Wi-Fi access point.

Device real-time data: An ATM generates data and communicates it to the server instantaneously through the Internet. This initiates and enables Online Transactions Processing (OLTP) in real time.

Event-driven device data: A device data can generate on an event only once. Examples are: (i) a device receive command from Controller or Monitor, and then performs action(s) using an actuator. When the action completes, then the device sends an acknowledgement; (ii) when an application seeks the status of a device, then the device communicates the status.

2. Data acquisition

Data acquisition means acquiring data from IoT or M2M devices. The data communicates after the interactions with a data acquisition system (application). The application interacts and communicates with a number of devices for acquiring the needed data. The devices send data on demand or at programmed intervals. Data of devices communicate using the network, transport and security layers.



An application can configure the devices for the data when devices have configuration capability. For example, the system can configure devices to send data at defined periodic intervals. Each device configuration controls the frequency of data generation. For example, system can configure an umbrella device to acquire weather data from the Internet weather service, once each working day in a week. An ACVM can be configured to communicate the sales data of machine and other information, every hour. The ACVM system can be configured to communicate instantaneously in event of fault or in case requirement of a specific chocolate flavour needs the Fill service.

Application can configure sending of data after filtering or enriching at the gateway at the data-adaptation layer. The gateway in-between application and the devices can provision for one or more of the following functions—transcoding, data management and device management. Data management may be provisioning of the privacy and security, and data integration, compaction and fusion

Device-management software provisions for device ID or address, activation, configuring (managing device parameters and settings), registering, deregistering, attaching, and detaching

3. Data validation

Data acquired from the devices does not mean that data are correct, meaningful or consistent. Data consistency means within expected range data or as per pattern or data not corrupted during transmission. Therefore, data needs validation checks. Data validation software do the validation checks on the acquired data. Validation software applies logic, rules and semantic annotations. The applications or services depend on valid data. Then only the analytics, predictions, prescriptions, diagnosis and decisions can be acceptable.

Large magnitude of data is acquired from a large number of devices, especially, from machines in industrial plants or embedded components data from large number of automobiles or health devices in ICUs or wireless sensor networks, and so on. Validation software, therefore, consumes significant resources. An appropriate strategy needs to be adopted. For example, the adopted strategy may be filtering out the invalid data at the gateway or at device itself or controlling the frequency of acquiring or cyclically scheduling the set of devices in industrial systems. Data enriches, aggregates, fuses or compacts at the adaptation layer.

4. Data Categorization for Storage

Services, business processes and business intelligence use data. Valid, useful and relevant data can be categorized into three categories for storage—data

alone, data as well as results of processing, only the results of data analytics are stored. Following are three cases for storage:

a. Data which needs to be repeatedly processed, referenced or audited in future, and therefore, data alone needs to be stored.

b. Data which needs processing only once, and the results are used at a later time using the analytics, and both the data and results of processing and analytics are stored. Advantages of this case are quick visualization and reports generation without reprocessing. Also the data is available for reference or auditing in future.

c. Online, real-time or streaming data need to be processed and the results of this processing and analysis need storage.

Data from large number of devices and sources categorizes into a fourth category called Big data. Data is stored in databases at a server or in a data warehouse or on a Cloud as Big data.

5. Assembly Software for the Events

A device can generate events. For example, a sensor can generate an event when temperature reaches a preset value or falls below a threshold. A pressure sensor in a boiler generates an event when pressure exceeds a critical value which warrants attention. Each event can be assigned an ID. A logic value sets or resets for an event state. Logic 1 refers to an event generated but not yet acted upon. Logic 0 refers to an event generated and acted upon or not yet generated. A software component in applications can assemble the events (logic value, event ID and device ID) and can also add Date time stamp. Events from IoTs and logic-flows assemble using software.

6. Data store

A data store is a data repository of a set of objects which integrate into the store. Features of data store are:

Objects in a data-store are modeled using Classes which are defined by the database schemas

A data store is a general concept. It includes data repositories such as database, relational database, flat file, spreadsheet, mail server, web server, directory services and VMware

A data store may be distributed over multiple nodes. Apache Cassandra is an example of distributed data store.

A data store may consist of multiple schemas or may consist of data in only one scheme. Example of only one scheme data store is a relational database.

Repository in English means a group, which can be related upon to look for required things, for special information or knowledge. For example, a repository

of paintings of artists. A database is a repository of data which can be relied upon for reporting,

analytics, process, knowledge discovery and intelligence. A flat file is another repository.

7. Data Centre Management

A data centre is a facility which has multiple banks of computers, servers, large memory systems, high speed network and Internet connectivity. The centre provides data security and protection using advanced tools, full data backups along with data recovery, redundant data communication connections and full system power as well as electricity supply backups.

Large industrial units, banks, railways, airlines and units for whom data are the critical components use the services of data centres. Data centres also possess a dust free, heating, ventilation and air conditioning (HVAC), cooling, humidification and dehumidification equipment, pressurisation system with a physically highly secure environment. The manager of data centre is responsible for all technical and IT issues, operations of computers and servers, data entries, data security, data quality control, network quality control and the management of the services and applications used for data processing.

8. Server Management

Server management means managing services, setup and maintenance of systems of all types associated with the server. A server needs to serve around the clock. Server management includes managing the following:

● Short reaction times when the system or network is down

● High security standards by routinely performing system maintenance and updation ● Periodic system updates for state-of-the art setups

● Optimised performance

● Monitoring of all critical services, with SMS and email notifications

● Security of systems and protection

● Maintaining confidentiality and privacy of data

● High degree of security and integrity and effective protection of data, files and databases at the organisation

● Protection of customer data or enterprise internal documents by attackers which includes spam mails, unauthorised use of the access to the server, viruses, malwares and worms

● Strict documentation and audit of all activities.

9. Spatial Storage

Consider goods with RFID tags. When goods move from one place to another, the IDs of goods as well as locations are needed in tracking or inventory control applications. Spatial storage is storage as spatial database which is optimised to store and later on receives queries from the applications. Suppose a digital map is required for parking slots in a city. Spatial data refers to data which represents objects defined in a geometric space. Points, lines and polygons are common geometric objects which can be represented in spatial databases. Spatial database can also represent database for 3D objects, topological coverage, linear networks, triangular irregular networks and other complex structures. Additional functionality in spatial databases enables efficient processing. Internet communication by RFIDs, ATMs, vehicles, ambulances, traffic lights, streetlights, waste containers are examples of where spatial database are used.

Spatial database functions optimally for spatial queries. A spatial database can perform typical SQL queries, such as select statements and performs a wide variety of spatial operations. Spatial database has the following features:

● Can perform geometry constructors. For example, creating new geometries

● Can define a shape using the vertices (points or nodes)

● Can perform observer functions using queries which replies specific spatial information such as location of the centre of a geometric object

● Can perform spatial measurements which mean computing distance between geometries, lengths of lines, areas of polygons and other parameters

● Can change the existing features to new ones using spatial functions and can predicate spatial relationships between geometries using true or false type queries.