SMU
SINGAPORE MANAGEMENT UNIVERSITY

School of
Computing and
Information Systems

Foundations of Cyber-Physical Systems

Agenda

- CPS Data lifecycle
- Governing CPS Data throughout the CPS data lifecycle
- Monitoring and diagnostics
- Software update and maintenance
- Define and classify CPS risk scenarios
- Implementing CPS security controls

CPS Data Key Characteristics

- Big Data:
  - Huge amounts of data are generated, capturing detailed aspects of the
    - processes where devices are involved.
- Heterogeneous Data:
  - It is itself highly heterogeneous, differing on sampling rate, quality of captured values, etc.
- Real-World Data:
  - Relates to real-world processes and is dependent on the environment they interact with.
- Real-Time Data:
  - CPS data is generated in real-time and overwhelmingly can be communicated also in a very timely manner
  - Business value depends on the real-time processing of the info they convey.

VOLUME
Huge amount of data

VERACITY
Inconsistencies and uncertainty in data

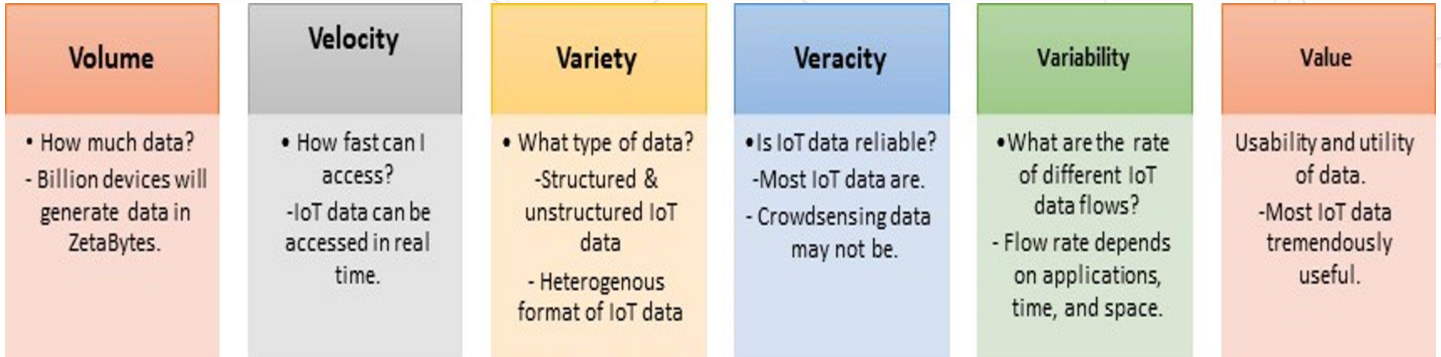
VARIETY
Different formats of data from various sources

VELOCITY
High speed of accumulation of data

VALUE
Extract useful data

Characteristics of big data

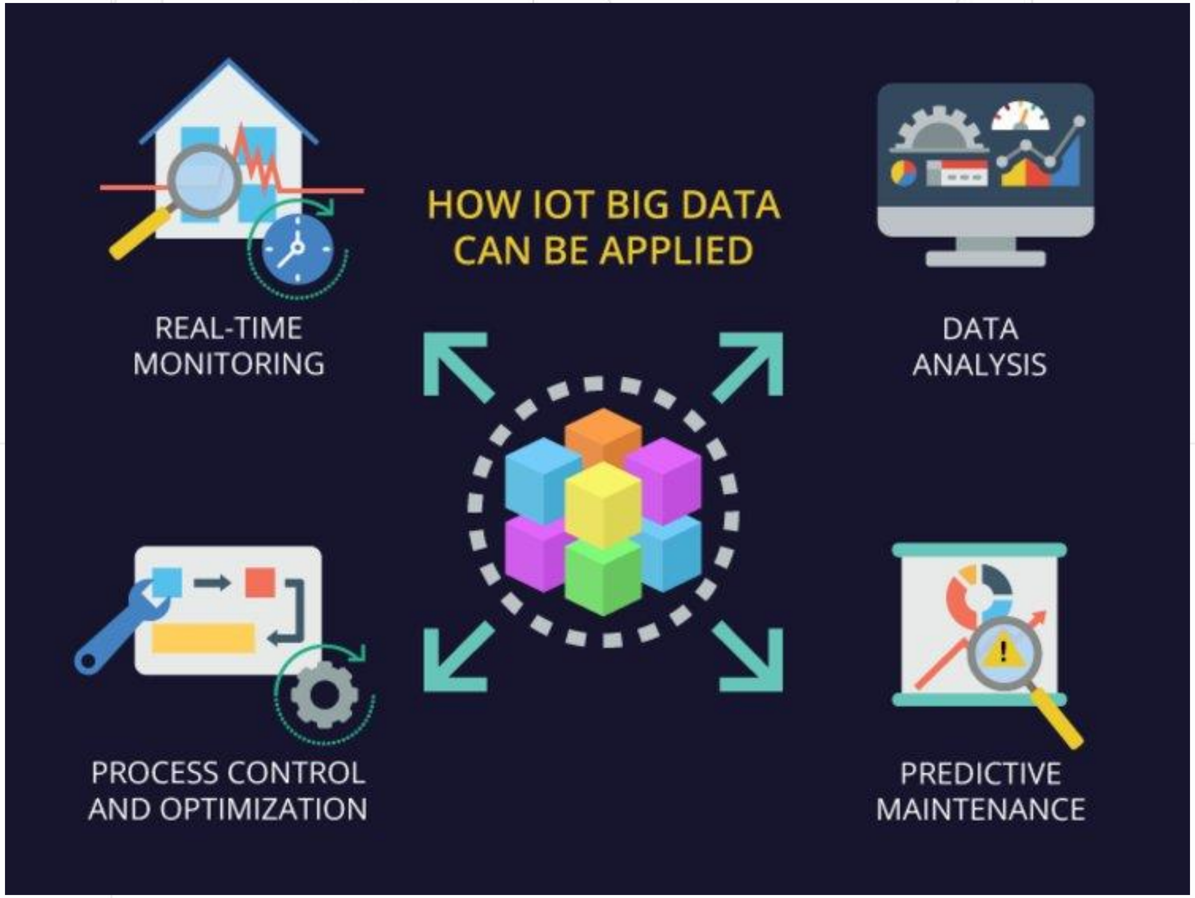
- Volume: amount of data
- Variety: data in different forms, from different sources
- Velocity: rate at which data flows
- Veracity: trustworthiness of the data
- Value: usable data



Volume	Velocity	Variety	Veracity	Variability	Value
• How much data? - Billion devices will generate data in ZetaBytes.	• How fast can I access? -IoT data can be accessed in real time.	• What type of data? -Structured & unstructured IoT data - Heterogenous format of IoT data	•Is IoT data reliable? -Most IoT data are. - Crowdsensing data may not be.	•What are the rate of different IoT data flows? - Flow rate depends on applications, time, and space.	Usability and utility of data. -Most IoT data tremendously useful.

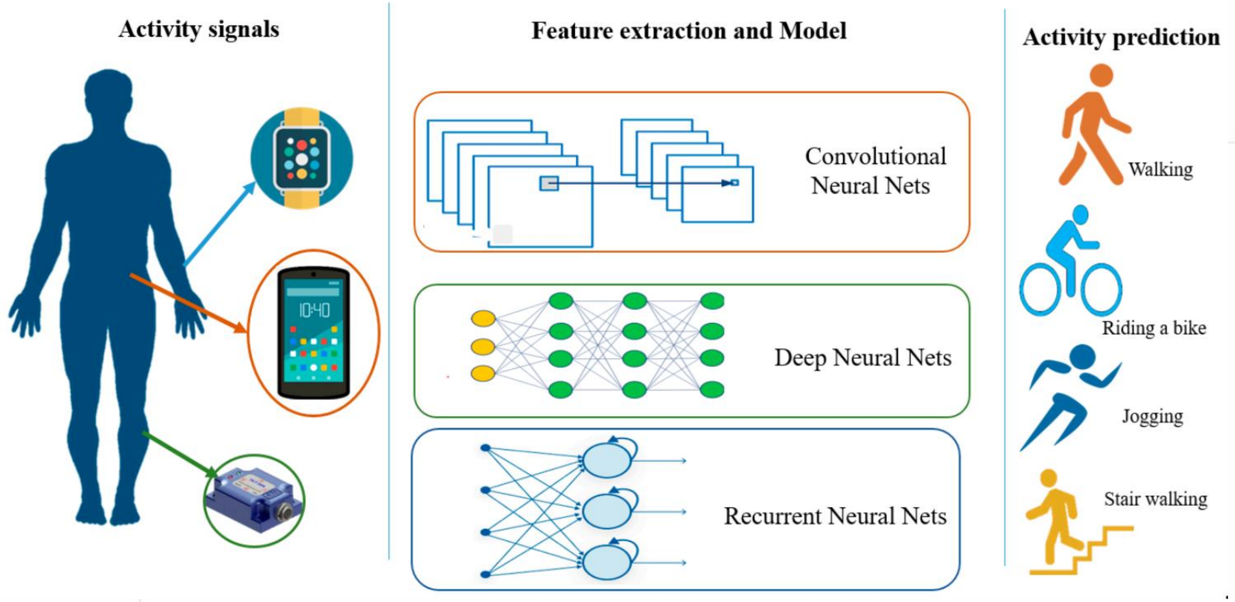
Big Data vs Cyber-Physical Data

- Human vs machine generated
- Historical vs real-time
- Non-streaming vs streaming
- Cloud analytics vs edge analytics

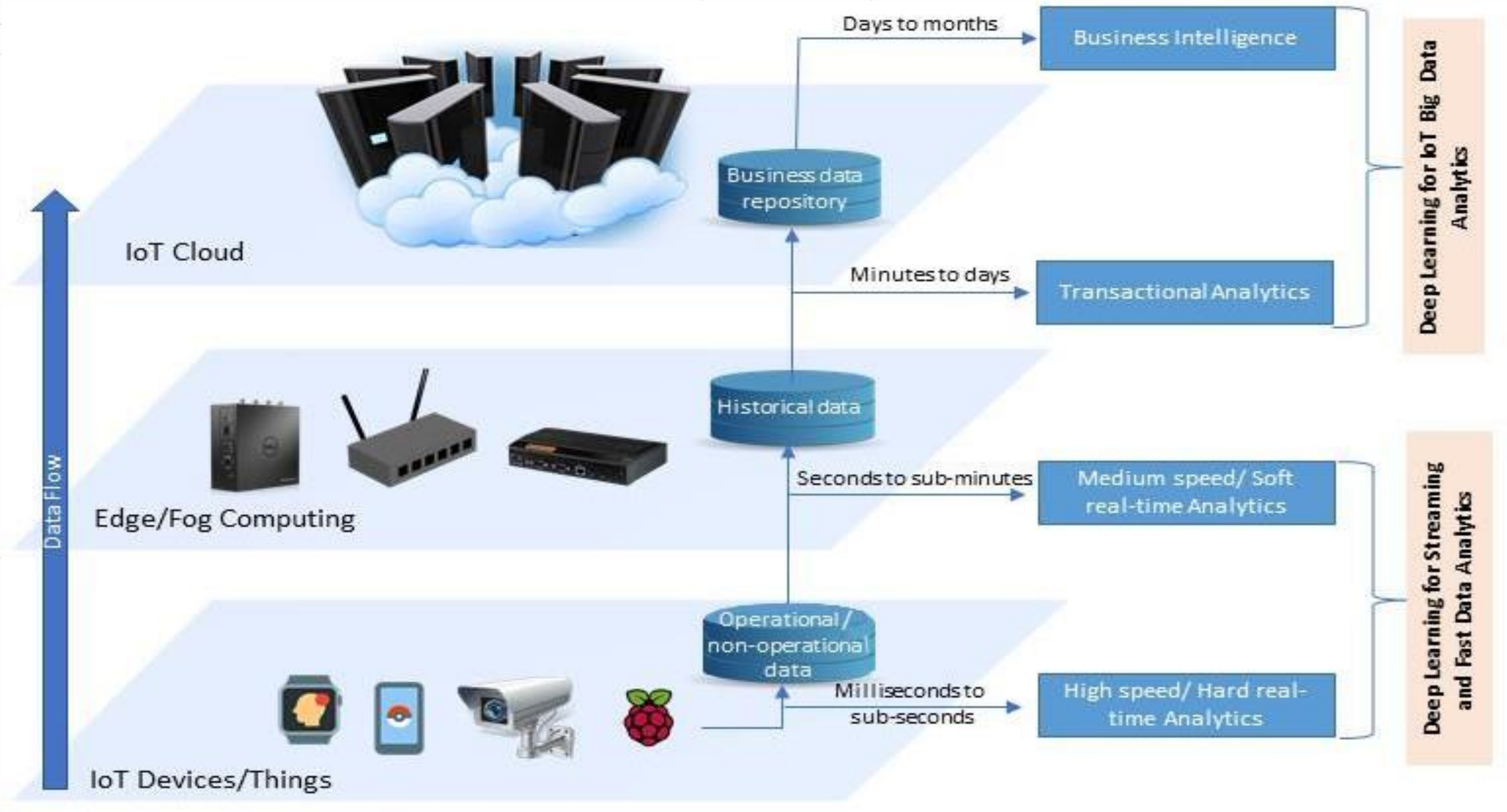

HOW IOT BIG DATA CAN BE APPLIED
REAL-TIME MONITORING
DATA ANALYSIS
PROCESS CONTROL AND OPTIMIZATION
PREDICTIVE MAINTENANCE

Sources of data

- Machine-generated data: data from environment, collected in the background in an unobtrusive way
- Crowd-sensed data: people willingly and intentionally sharing data
- Open data, a.k.a Open Source Intelligence: data in the public domain, usually from
    - Open data sites https://data.gov.sg/
    - Social media



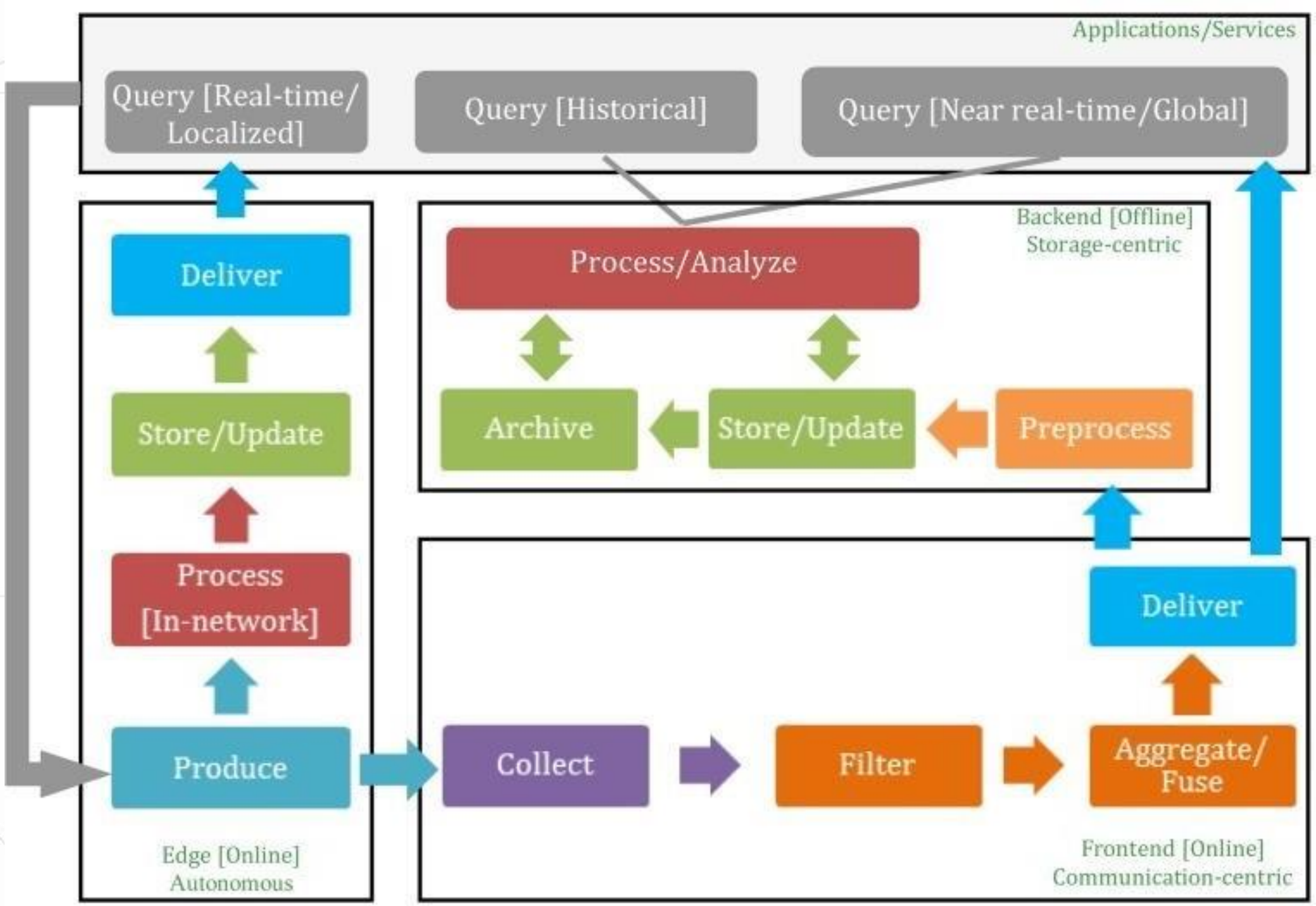
Key Characteristics of CPS Data and Its Analytics Requirements


Days to months
Business Intelligence
Business data repository
IoT Cloud
Minutes to days
Transactional Analytics
Deep Learning for IoT Big Data Analytics
Historical data
Edge/Fog Computing
Seconds to sub-minutes
Medium speed/ Soft real-time Analytics
Data Flow
Operational/ non-operational data
Milliseconds to sub-seconds
High speed/ Hard real-time Analytics
IoT Devices/Things
Deep Learning for Streaming and Fast Data Analytics

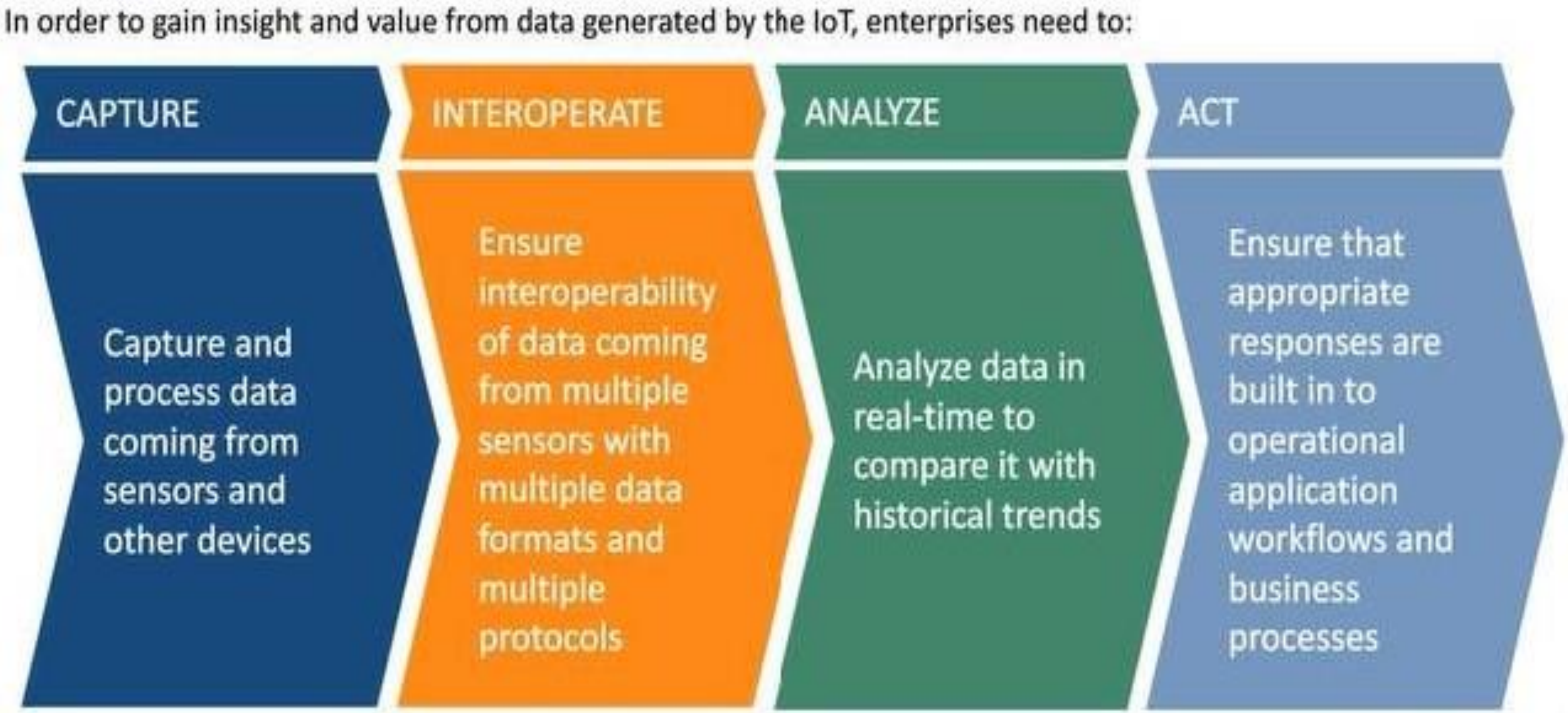
CPS Data Lifecycle

- The following are the key elements in the CPS data lifecycle
  - Querying
  - Production
  - Collection
  - Aggregation/Fusion
  - Delivery
  - Preprocessing
  - Storage /Update-Archiving
  - Processing /Analysis

CPS Data Lifecycle



CPS Data Processing Requirements



In order to gain insight and value from data generated by the IoT, enterprises need to:

CAPTURE

Capture and process data coming from sensors and other devices

INTEROPERATE

Ensure interoperability of data coming from multiple sensors with multiple data formats and multiple protocols

ANALYZE

Analyze data in real-time to compare it with historical trends

ACT

Ensure that appropriate responses are built in to operational application workflows and business processes

Data Management Framework

- Layered, data-centric, and federated paradigm
- “Things” layer
  - All entities and subsystems that can generate data
- Data repositories
  - Organizations or public
  - Specialized servers on the cloud
- Real-time or context-aware queries
- Discovery and engagement of data sources

CPS Data Management Requirements

- Managing data from CPS devices is an important aspect of a real-time analytics journey

- Versatile connectivity and ability to handle data variety
  - Data management system must be able to connect to all types of systems and various protocols so you can ingest data from those systems
  - Solution should support both structured and unstructured data

- Edge processing and enrichments
  - Should be able to filter out erroneous records coming from the CPS systems such as negative temperature readings—before ingesting it into the data lake
  - Should be able to enrich the data with metadata (such as timestamp or static text) to support better analytics

CPS Data Management Requirements

- Big data processing and machine learning
  - Performing real-time analytics
  - Anomaly detection should be possible in real time so that preventive
    - steps can be taken before it is too late

- Address data drift
  - Data coming from CPS systems can change over time due to events such as firm ware upgrades (data drift or schema drift)
  - Data management solution should automatically address data drift without interru pting the data management process

CPS Data Management Requirements

- Real-time monitoring and alerting
  - Should provide realtime monitoring with flow visualizations to show the status of the process at any time with respect to performance and throughput
  - Should also provide alerts for any issues
- Scalability and Agility
- Security

CPS Database Requirements

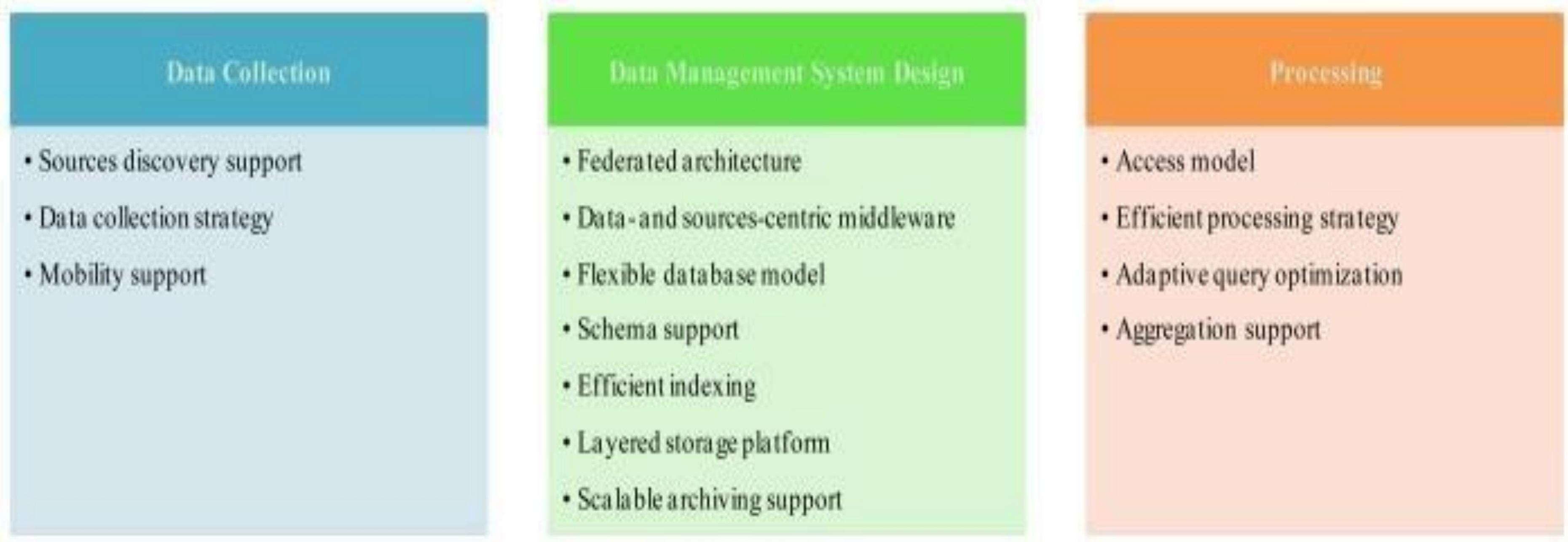
- **Scalability:** A database for CPS applications must be scalable

- **Fault tolerance:** An CPS database should also be fault tolerant and highly available

- **High availability**:
  Ensure high availability with regards to writes by using a distributed messaging system such as Apache Kafka or Amazon Kinesis, which is based on Apache Kafka

- **Flexibility**: CPS databases should be as flexible as required by the application.

Design Primitives for Comprehensive CPS Data Management Solution



Data Collection

- Sources discovery support
- Data collection strategy
- Mobility support

Data Management System Design

- Federated architecture
- Data-and sources-centric middleware
- Flexible database model
- Schema support
- Efficient indexing
- Layered storage platform
- Scalable archiving support

Processing

- Access model
- Efficient processing strategy
- Adaptive query optimization
- Aggregation support

CPS Data Management Strategy

- How much data are you going to capture on the Edge?
- How much data are you sending and transmitting to the cloud?
- What are you going to do with your data?
- How long are you going to keep it?
- Should you archive it when you no longer need it?

Data Security Issues

- Distributed frameworks
  - Most big data implementations actually distribute huge processing jobs across many systems for faster analysis
  - Hadoop is a well-known instance of open source tech involved in this, and originally had no security of any sort
  - Distributed processing may mean less data processed by any one system, but it means a l ot more systems where security issues can crop up
- Non-relational data stores
  - NoSQL databases, which by themselves usually lack security (which is instead provided, sort of, via middleware)

Data Security Issues

- Storage
  - In big data architecture, the data is usually stored on multiple tiers, depending on business needs for performance vs. cost
  - For instance, highpriority “hot” data will usually be stored on flash media. So locking down stor age will mean creating a tier-conscious strategy
- Endpoints
  - Security solutions that draw logs from endpoints will need to validate the authenticity of those e ndpoints, or the analysis isn’t going to do much good

Data Security Issues

- Real-time security/compliance
  - The key is finding a way to ignore the false positives

- Data mining
  - Find the patterns that suggest business strategies.
  - Ensure they're secured against not just external threats, but insiders who abuse network privileges to obtain sensitive information – yet another layer of big data security issues

Data Security Issues

- Access controls
  - Just as with enterprise IT as a whole, it’s critically important to provide
    - a system in which encrypted authentication/validation

- Granular auditing
  - can help determine when missed attacks haveoccurred, what the consequences were, and w hat should be done to improve matters in the future

Data Security Issues

- Data provenance/Data Lineage
  - Primarily concerns metadata (data about data), which can be extremely helpful in determining where data came from, who accessed it, or what was done with it.
  - Usually, this kind of data should be analyzed with exceptional speed to minimize the time in whi ch a breach is active.
  - Privileged users engaged in this type of activity must be thoroughly vetted and closely monitore d to ensure they don’t become their own big data security issues.

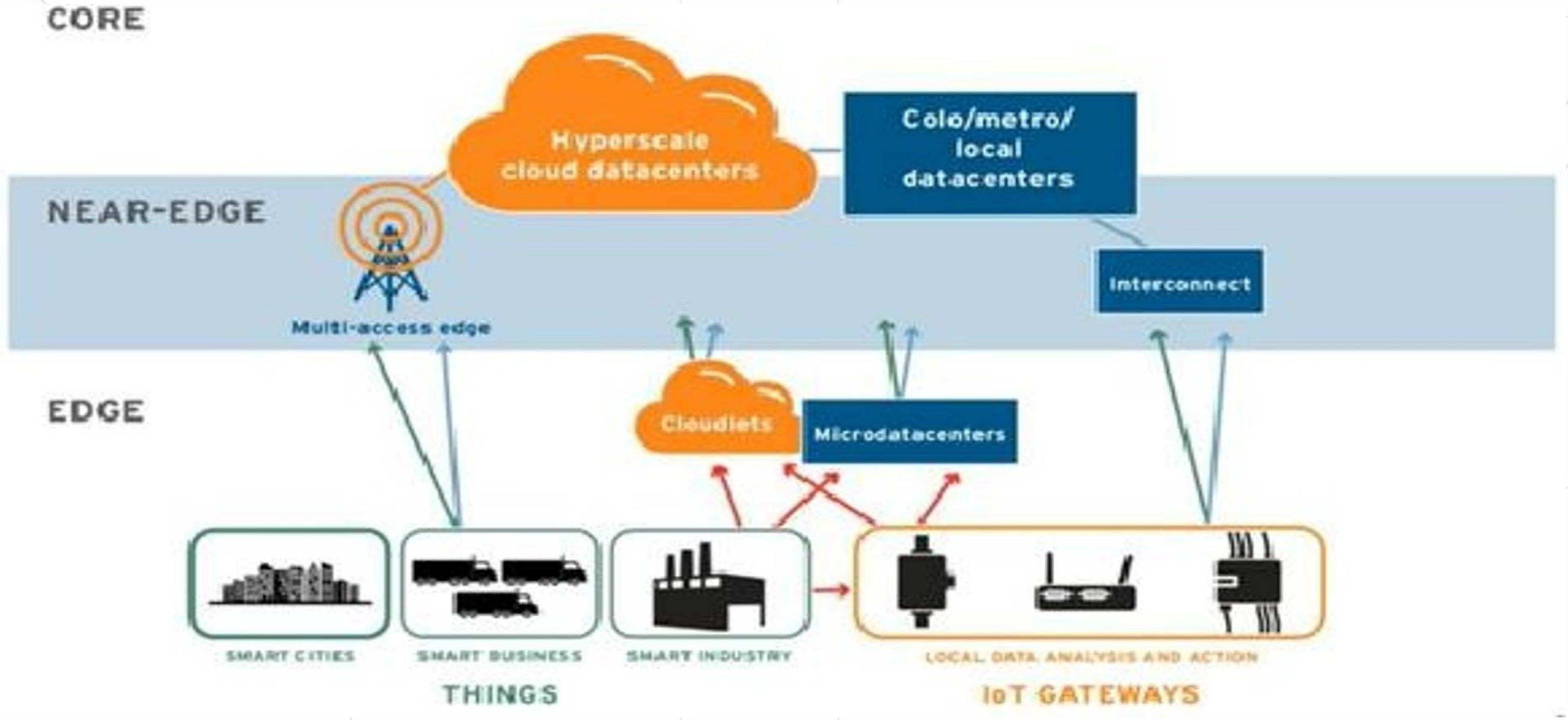
Security Measures for CPS Data

- Encryption
- Centralized Key Management
- User Access Control
- Intrusion Detection and Prevention
- Physical Security

Scalability

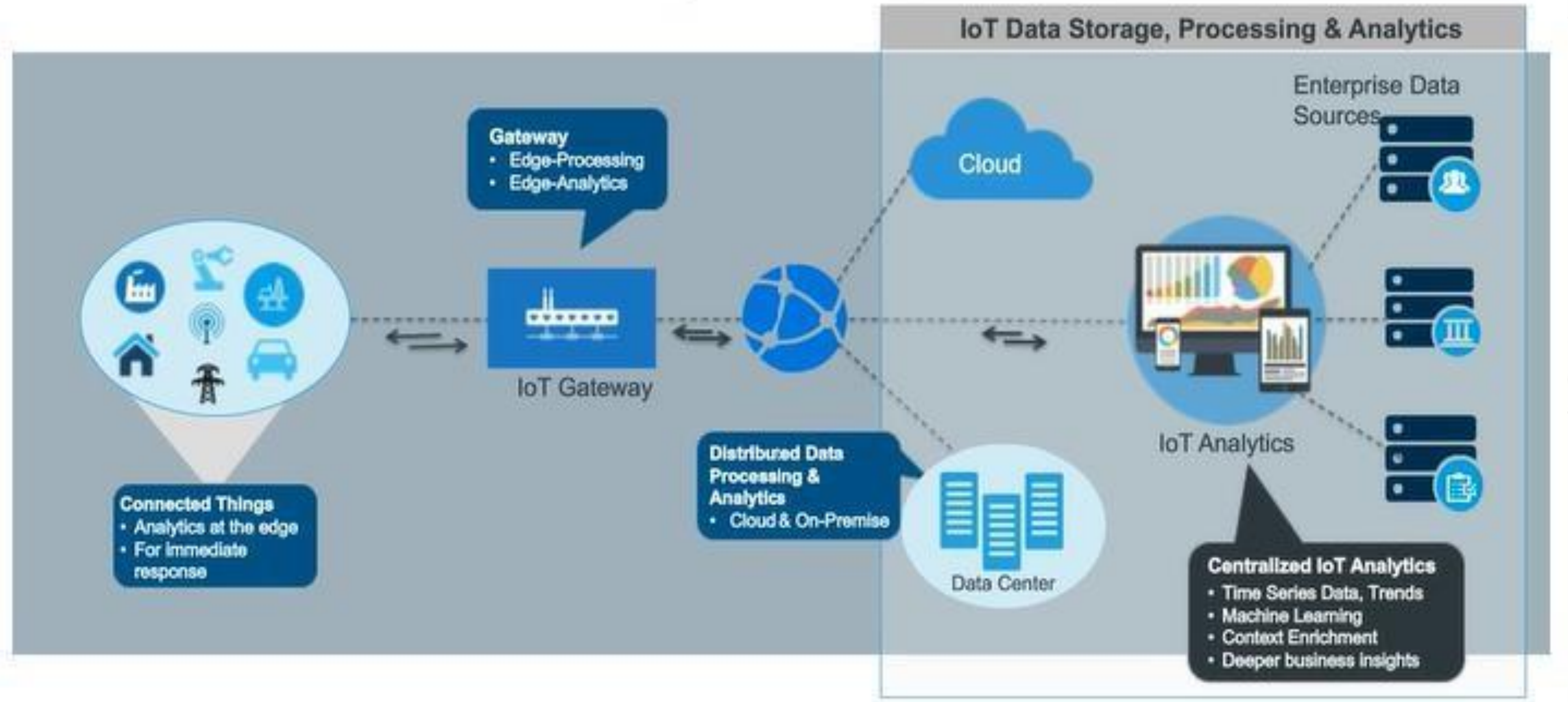
Ability to secure multiple types of data in different stages.

High-Level CPS Data Architecture



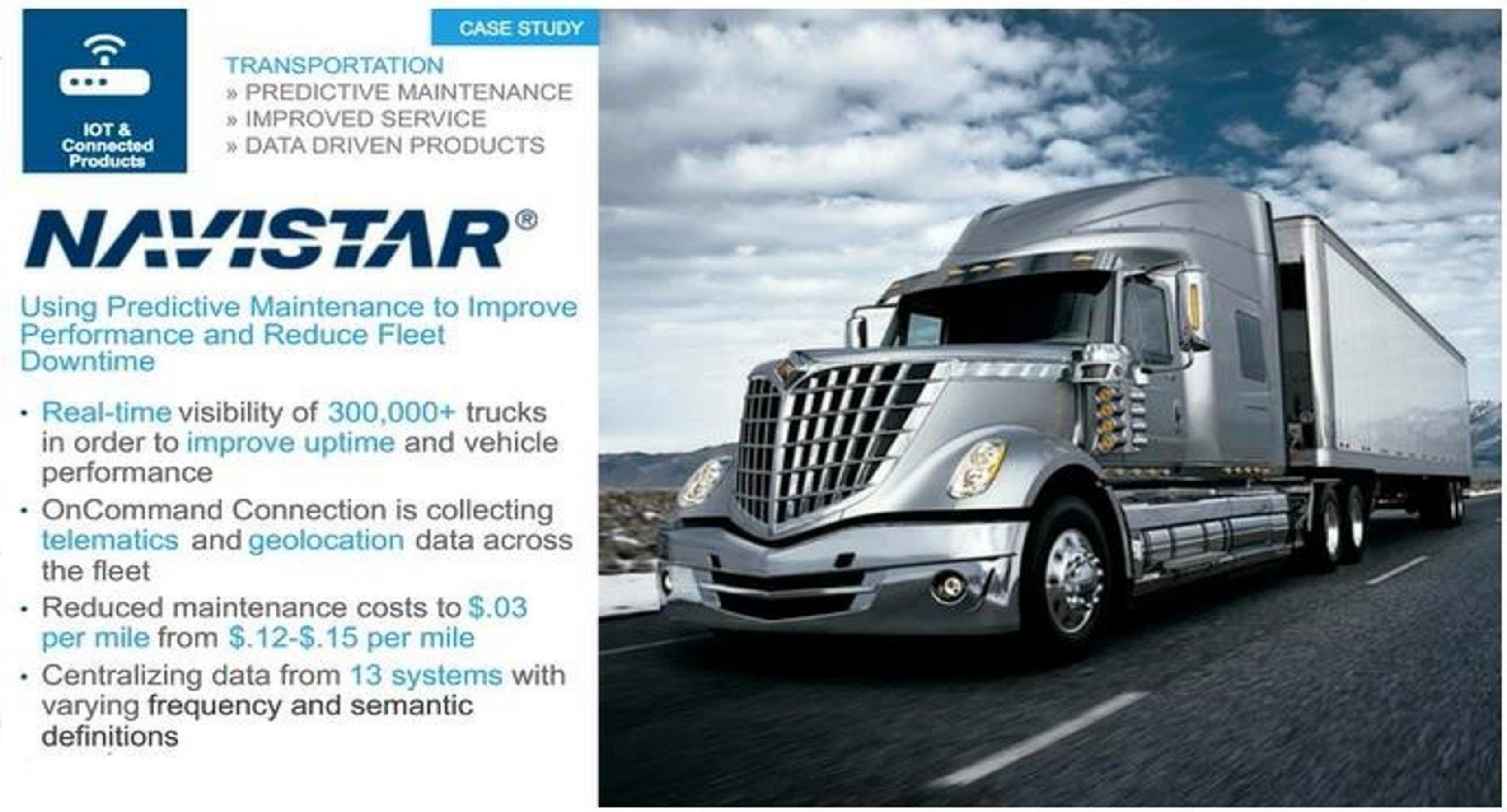
CPS Analytics Continuum – Edge, Near Edge, Cloud

The ecosystem and architecture



USE CASE

Use Case



IOT & Connected Products

CASE STUDY

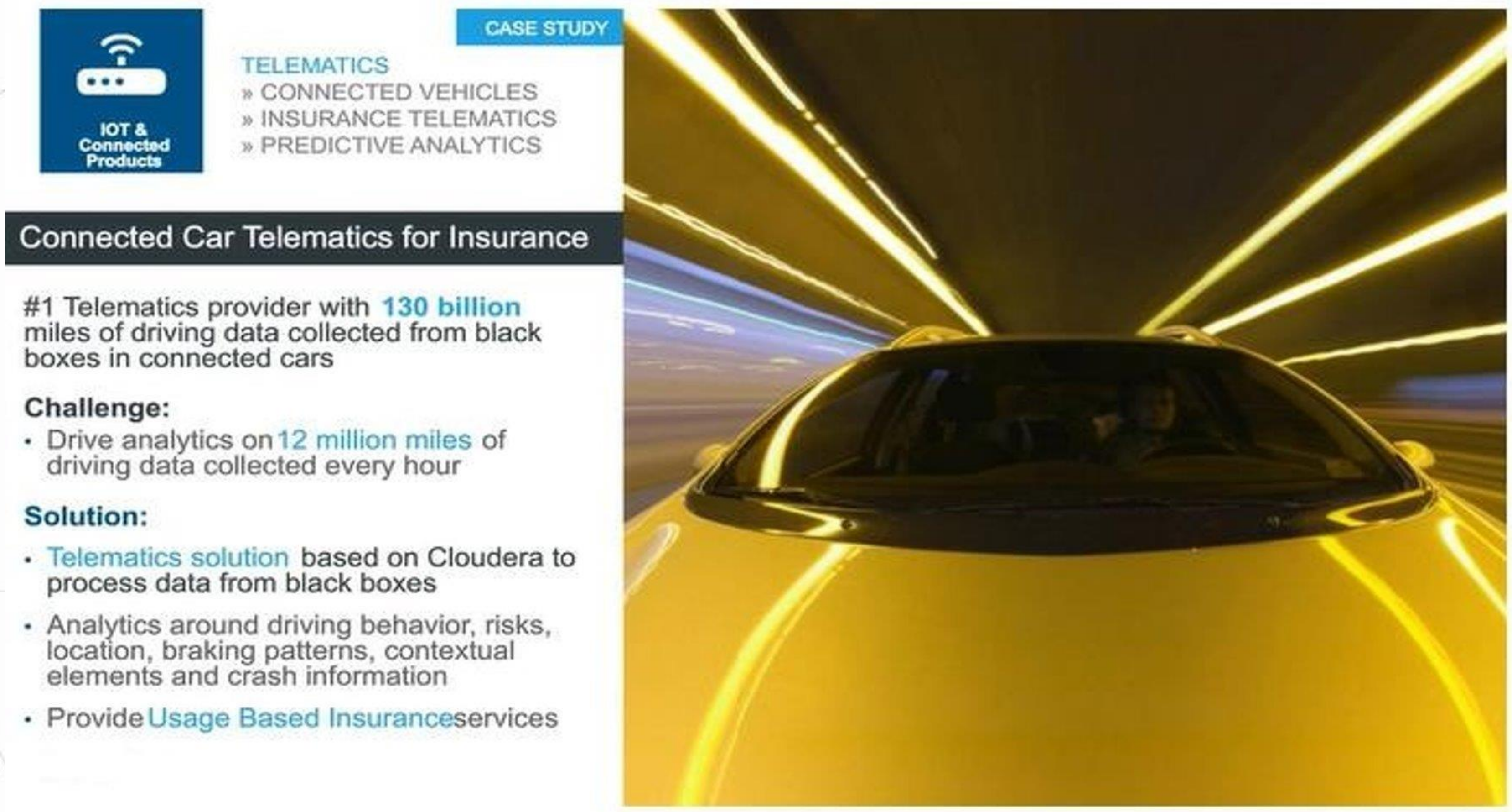
TRANSPORTATION
» PREDICTIVE MAINTENANCE
» IMPROVED SERVICE
» DATA DRIVEN PRODUCTS

NAVISTAR®

Using Predictive Maintenance to Improve Performance and Reduce Fleet Downtime

- Real-time visibility of 300,000+ trucks in order to improve uptime and vehicle performance
- OnCommand Connection is collecting telematics and geolocation data across the fleet
- Reduced maintenance costs to $.03 per mile from $.12-$.15 per mile
- Centralizing data from 13 systems with varying frequency and semantic definitions

Use Case



IOT & Connected Products

CASE STUDY

TELEMATICS
» CONNECTED VEHICLES
» INSURANCE TELEMATICS
» PREDICTIVE ANALYTICS

Connected Car Telematics for Insurance

#1 Telematics provider with **130 billion** miles of driving data collected from black boxes in connected cars

Challenge:

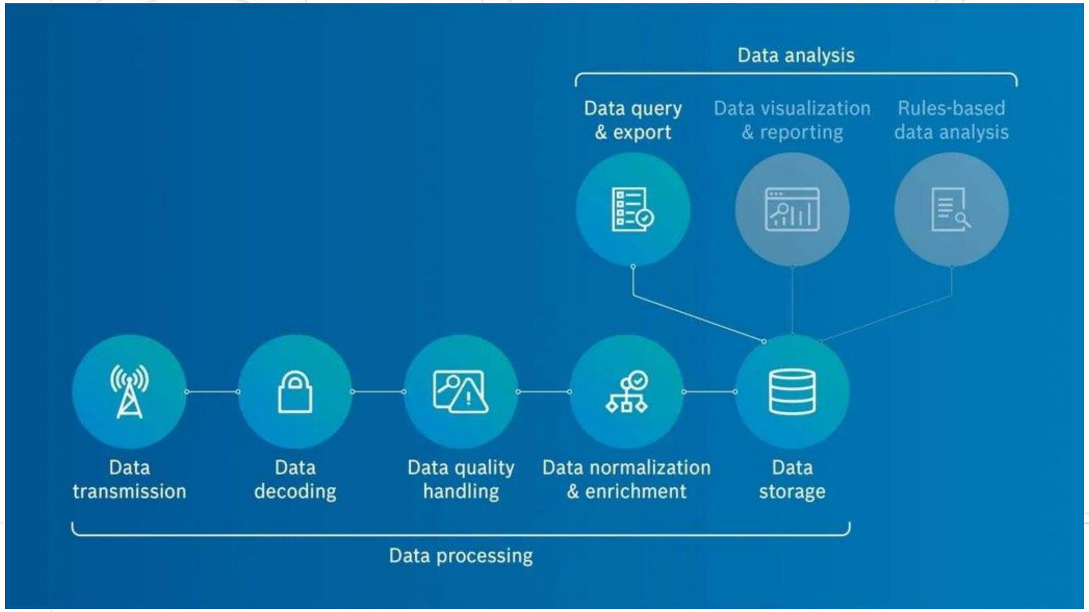
- Drive analytics on 12 million miles of driving data collected every hour

Solution:

- Telematics solution based on Cloudera to process data from black boxes
- Analytics around driving behavior, risks, location, braking patterns, contextual elements and crash information
- Provide Usage Based Insuranceservices

Benefits of CPS Data Management

- Understand users' needs
- Predict asset wear
- Enable resource efficiency
- Create effective systems



CPS & Business Intelligence

- The value of CPS comes to live when we are able to harness the power of data

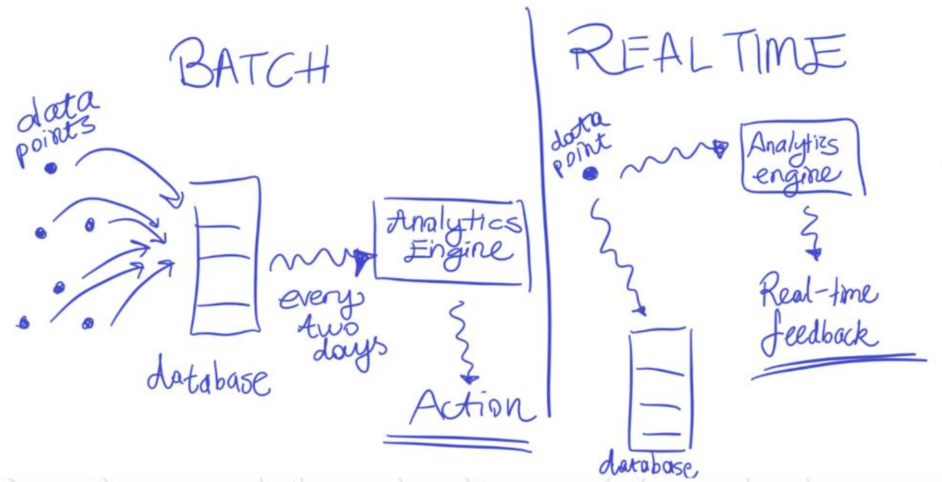
Edge Computing

- Perform computations nearer to the edge of the network, rather than in the cloud
- Motivation:
  - Bandwidth issues
  - Connectivity issues
- Use cases:
  - On-site business intelligence
  - Autonomous vehicles
  - Industrial plants

Energy-aware computation

- Dynamic Voltage and Frequency Scaling (DVFS): adjust the operating voltage and frequency of the processor based on workload requirements
- Computational offloading: distribute the computational workload among different components or devices in the system, including the cloud
- Collaborative computation: multiple devices within a networked Cyber-Physical System work together to share the computational workload

Timeliness



- Real-time data
- Data arrives as a stream
- Can be stored in the cloud, or on the devices
- Collected and processed as soon as possible
- Usually used for event detection
  - Reactive: detect events after they have happened
  - Predictive: detect events before they happen
- Replication, caching: in practice, having too much data can cause database queries to be slow, which affects real-time requirements. To improve performance, maintain a cached replica which caches data over a rolling time window - last few minutes, hours, days, etc
- Stream analytics, real-time analytics

Analytics Processes

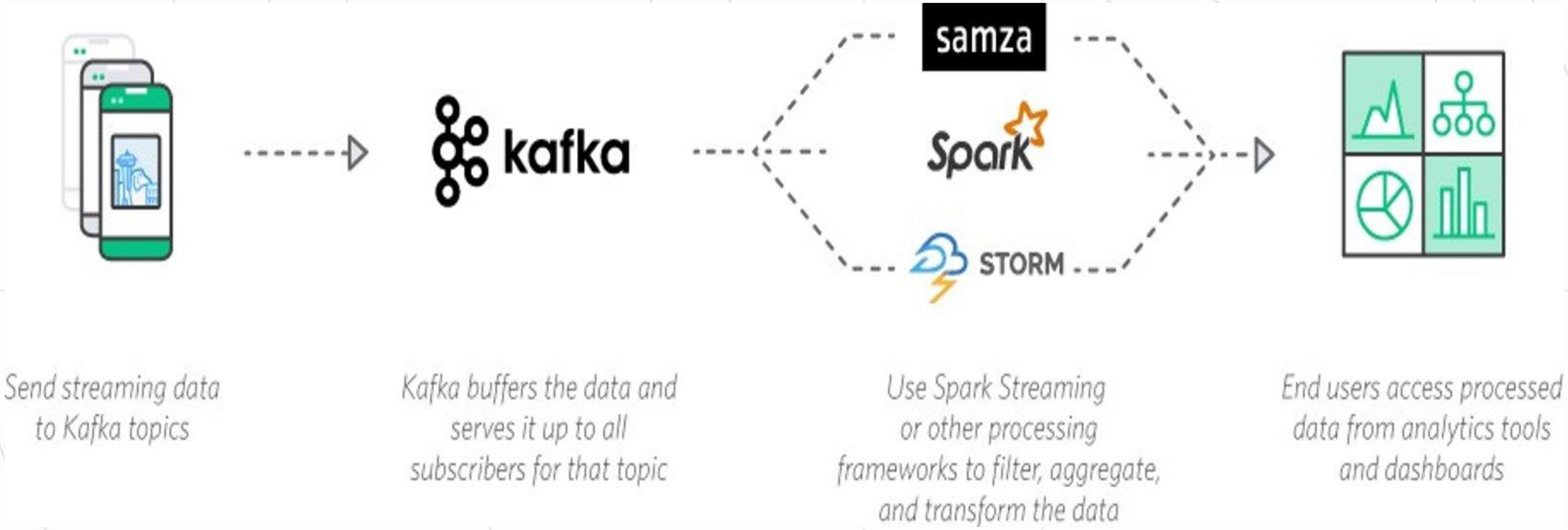
- Collect: get data from various sources
- Prepare
    - Clean
        - Remove invalid data: duplicate data, unwanted data
        - Fix structural errors: inconsistent data entry inputs, typos, different letter cases for the same value
        - Filter outliers: data that seems out of place, exceeds sensor range
        - Handle missing data: drop data points, reconstruct, or make analysis work with missing values
    - Transform: convert from raw data into another form
        - Unit conversion, e.g. Celsius vs Fahrenheit
        - Formats, e.g. json, xml
        - Structure, e.g. queues, graphs, linked lists, hashmaps
        - Moving window statistics

Analytics Processes

- Enrich: fuse data sources together
- Analyze
  - Explore, focus. Diverge, converge.
  - Ask questions about the data, derive the answers
    - Real-time, reactive
    - Real-time, predictive
    - Historical
- Look for patterns, unexpected insights. Birds example
- Apply:
  - Send notifications to a human person
  - Control some actuators
  - Make data-driven decisions

Data Collectors

- Kafka: Is used for develop real time data pipelines and streaming CPS architecture It is horizontally scalable, fault- tolerant, wicked fast.



Send streaming data to Kafka topics

Kafka buffers the data and serves it up to all subscribers for that topic

Use Spark Streaming or other processing frameworks to filter, aggregate, and transform the data

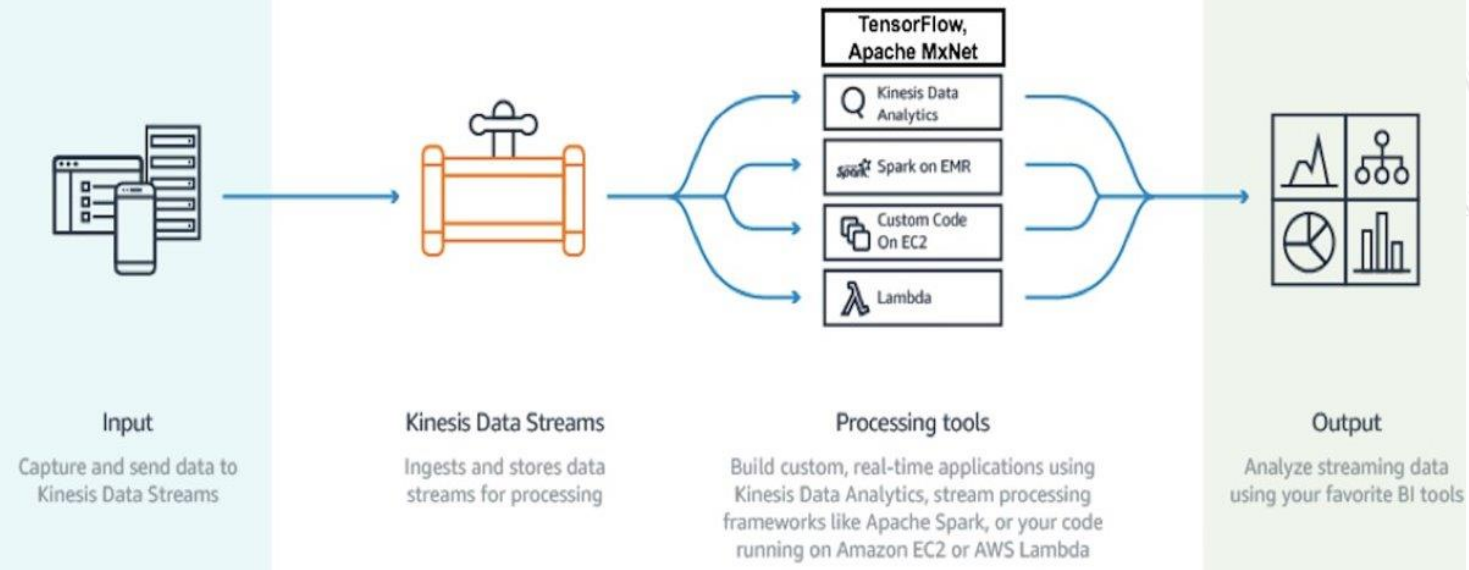
End users access processed data from analytics tools and dashboards

Data Collectors

- Apache Flume: Is a distributed, reliable, and available software for efficiently collecting, aggregating, and moving large amounts of log data.

Data Collectors

- Amazon Kinesis collects, process, and analyze real-time, streaming data in cost-effective way, so you can get timely insights and react quickly to new input.
- Can ingest real time data such as audio, video, website clickstreams, application logs, and CPS telemetry data for machine learning, analytics, and other applications.



Capture and send data to Kinesis Data Streams

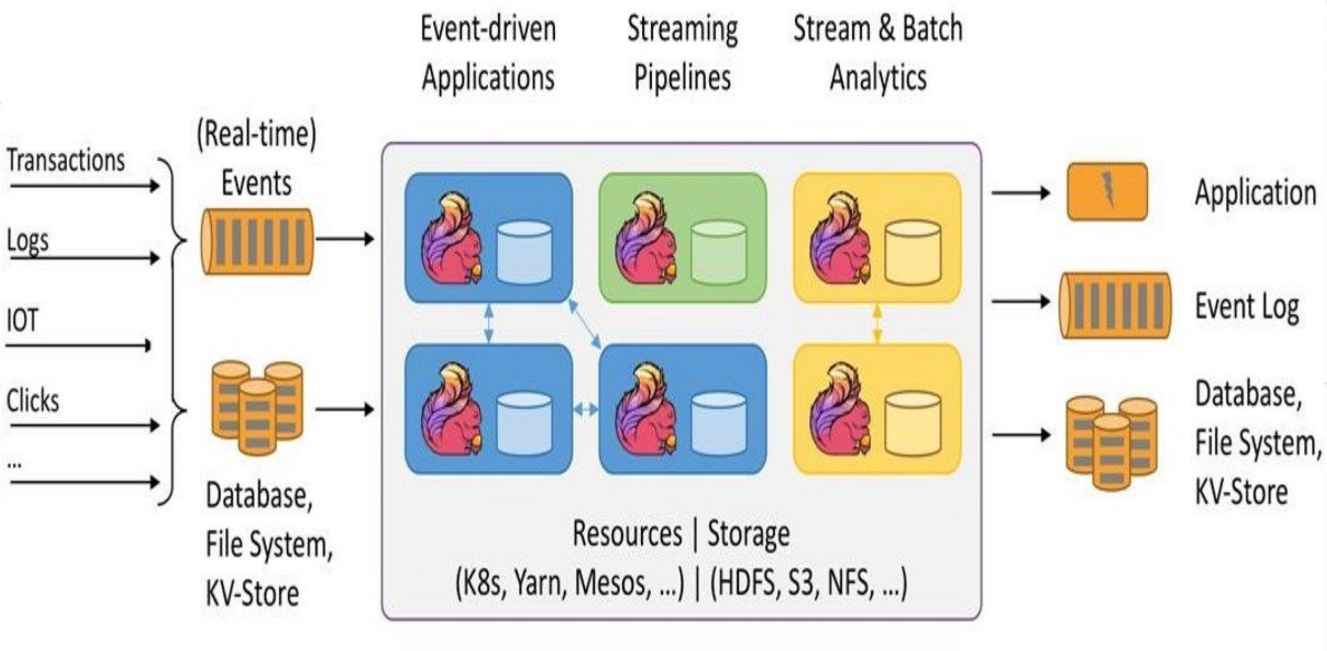
Ingests and stores data streams for processing

Build custom, real-time applications using Kinesis Data Analytics, stream processing frameworks like Apache Spark, or your code running on Amazon EC2 or AWS Lambda

Analyze streaming data using your favorite BI tools

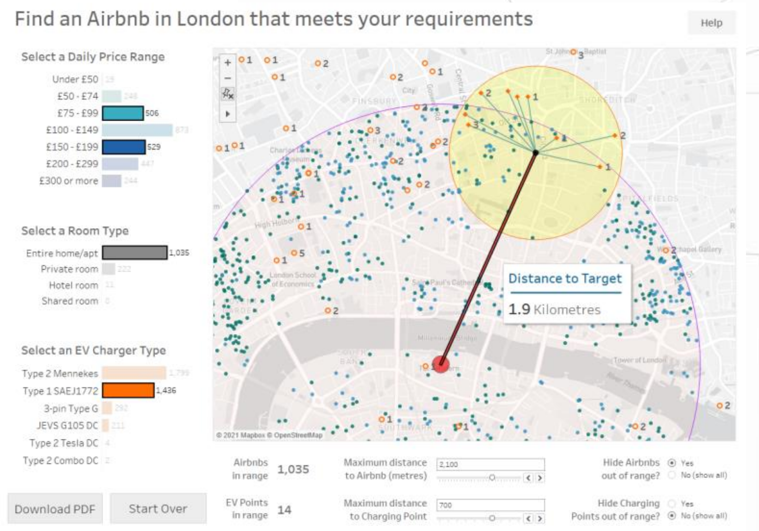
Data Processing

- This layer is responsible for processing CPS big data and analyzing the device informatio n to extract insights from the massive data set. Following are some “Big Data” platforms fo r real-time data processing.
  - Apache Spark
  - Apache Storm
  - S4
  - Amazon Kinesis
  - Microsoft StreamInsight
  - Apache Flink


Event-driven Applications
Streaming Pipelines
Stream & Batch Analytics
Transactions
Logs
IOT
Clicks
...
(Real-time) Events
Database, File System, KV-Store
Resources | Storage
(K8s, Yarn, Mesos, ...) | (HDFS, S3, NFS, ...)
Application
Event Log
Database, File System, KV-Store

Analytics

- Descriptive analytics — “what happened?”
- Diagnostic analytics — “why did it happen?”
- Predictive analytics — “what might happen in the future?”
- Prescriptive analytics — “what are the best actions to take?”



Find an Airbnb in London that meets your requirements

Select a Daily Price Range

Under £50	15
£50 - £74	248
£75 - £99	506
£100 - £149	873
£150 - £199	529
£200 - £299	447
£300 or more	244

Select a Room Type

Entire home/apt	1,035
Private room	222
Hotel room	11
Shared room	0

Select an EV Charger Type

Type 2 Mennekes	1,799
Type 1 SAEJ1772	1,436
3-pin Type G	292
JEVS G105 DC	211
Type 2 Tesla DC	4
Type 2 Combo DC	2

Distance to Target 1.9 Kilometres

Airbnbs in range 1,035

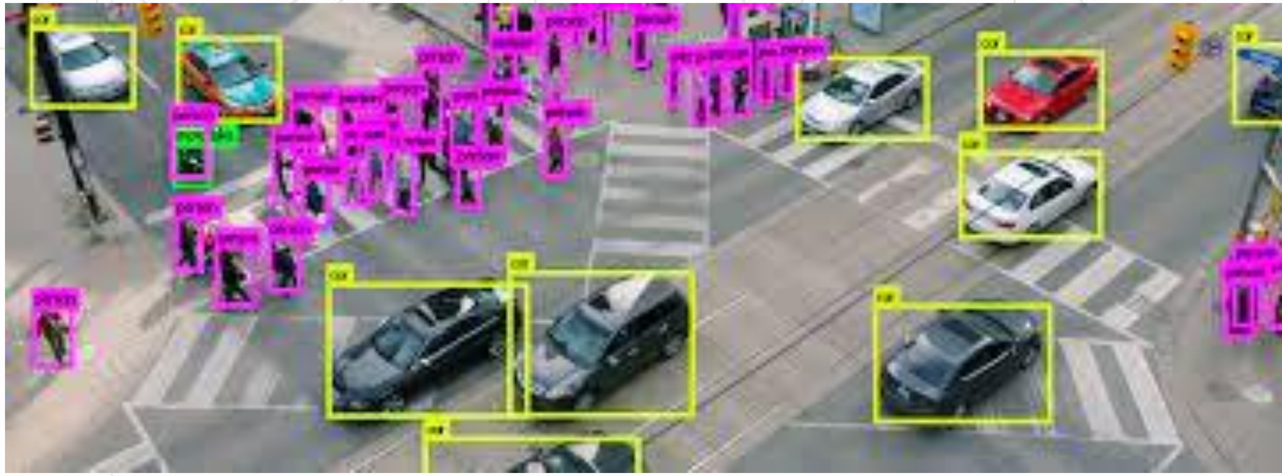
Maximum distance to Airbnb (metres) 2,100

Hide Airbnbs out of range? ● Yes ○ No (show all)

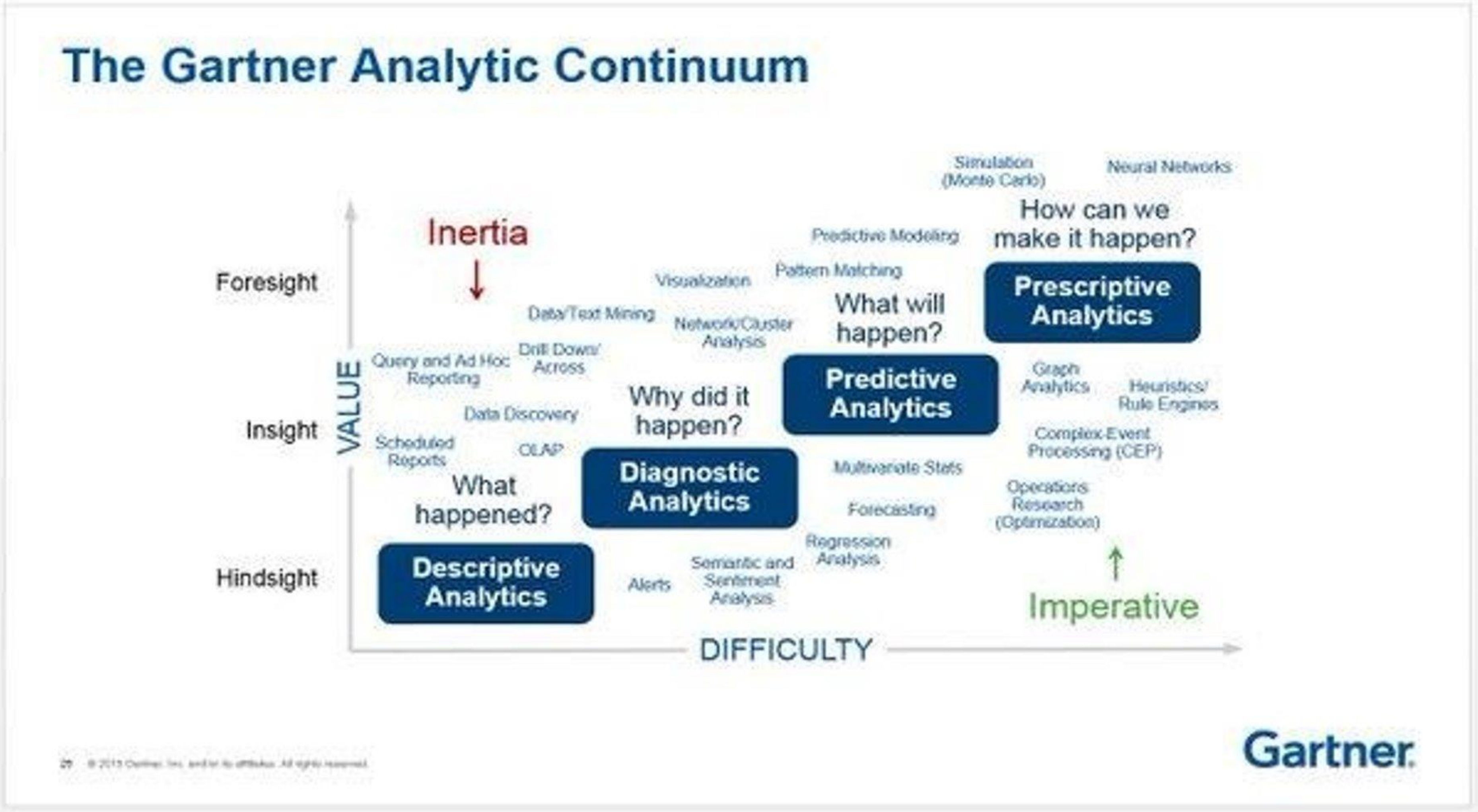
EV Points in range 14

Maximum distance to Charging Point 700

Hide Charging Points out of range? ○ Yes ● No (show all)



Analytics



The Gartner Analytic Continuum

Foresight

Insight

Hindsight

VALUE

DIFFICULTY

Inertia

Imperative

What happened?

Descriptive Analytics

Why did it happen?

Diagnostic Analytics

What will happen?

Predictive Analytics

How can we make it happen?

Prescriptive Analytics

Query and Ad Hoc Reporting

Scheduled Reports

Data Discovery

OLAP

Drill Down/ Across

Data/Text Mining

Visualization

Network/Cluster Analysis

Alerts

Semantic and Sentiment Analysis

Regression Analysis

Forecasting

Multivariate Stats

Pattern Matching

Predictive Modeling

Simulation (Monte Carlo)

Neural Networks

Graph Analytics

Heuristics/ Rule Engines

Complex Event Processing (CEP)

Operations Research (Optimization)

Gartner

29 © 2015 Gartner, Inc. and/or its affiliates. All rights reserved.

Privacy-preserving analytics

- Analyze data in a way that protects the privacy and confidentiality of the persons involved in the data
- Some data from Cyber-Physical Systems could be sensitive or personally identifiable
- Data anonymization
- Data aggregation
- Differential privacy
- Federated learning

Cyber-Physical Clouds

- Cyber-Physical clouds integrate Cyber-Physical Systems with cloud computing
- Computational power: shift intensive computations from Cyber-Physical Systems devices to much more powerful machines in the cloud
- Sensor cloud virtualization
    - interoperability in Cyber-Physical Systems is challenging due to their heterogeneous nature.
    - Resources such as sensors and actuators could be virtualized.
    - Multiple users share several physical sensors seamlessly through common interfaces, using suitable abstraction layers on top of physical sensor devices.

Considerations

- Processing speed must be fast enough for data velocity
- Latency: after processing data, how fast must the response be: seconds, minutes, hours?
- Storage capacity must be large enough for data volume
- Data to store: raw, semi-processed, processed, actionable, etc

Predictive Maintenance – Use Case

Setting	Use Cases	Customer Case Study—Description
Automotive	**Predictive Maintenance – Connected Vehicles**	One of the leading auto manufacturers in North America is using Cloudera as its data management platform to monitor the health of 300,000+ trucks in real time in order to improve uptime and reduce fleet maintenance costs by 40 percent.
Manufacturing	**Predictive Maintenance – Industrial CPS**	A leading industrial automation and robotics company is utilizing Cloudera to ingest, store, and analyze streaming sensor data from thousands of industrial robots, in real time, in order to eliminate machine downtime.
Heavy Machinery	**Predictive Maintenance – Heavy Machinery**	One of the biggest heavy equipment fleet manufacturers in North America is using Cloudera to parse large-volume and high-velocity data from sensors to continuously monitor performance of

CPS Analytics in a nutshell

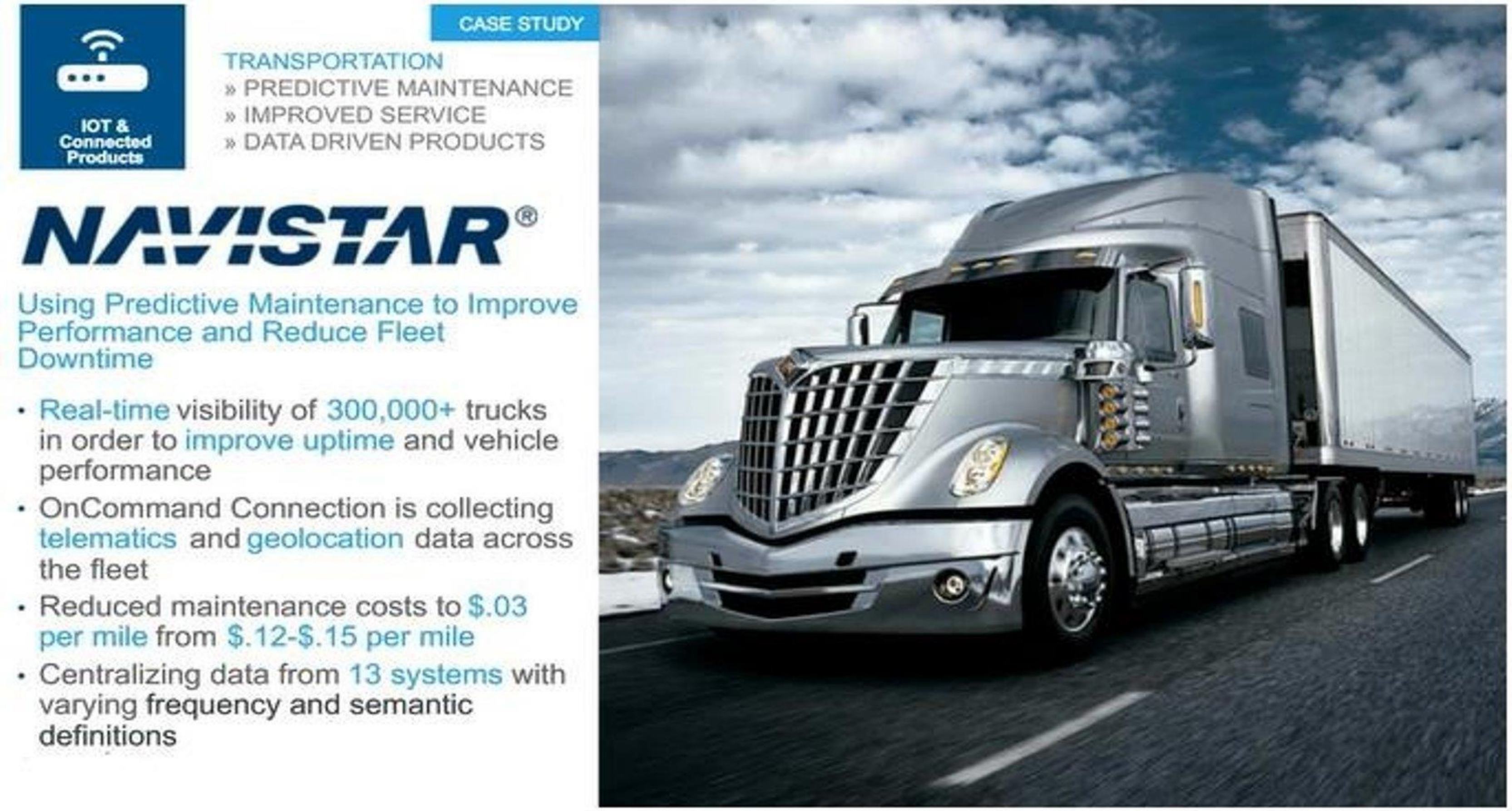
- Descriptive & Diagnostic Analytics (backward looking)
- Predictive Analytics (forward looking)
- Prescriptive Analytics (Next Best Action)

● There is no “one-size-fits-all” solution at the intersection of

● business needs and advanced analytics.

● Knowledge, expertise and the current technologies evolution enable the availability of open source predictive analytics as a service approach.

Use Case



CASE STUDY

IOT & Connected Products

TRANSPORTATION
» PREDICTIVE MAINTENANCE
» IMPROVED SERVICE
» DATA DRIVEN PRODUCTS

NAVISTAR®

Using Predictive Maintenance to Improve Performance and Reduce Fleet Downtime

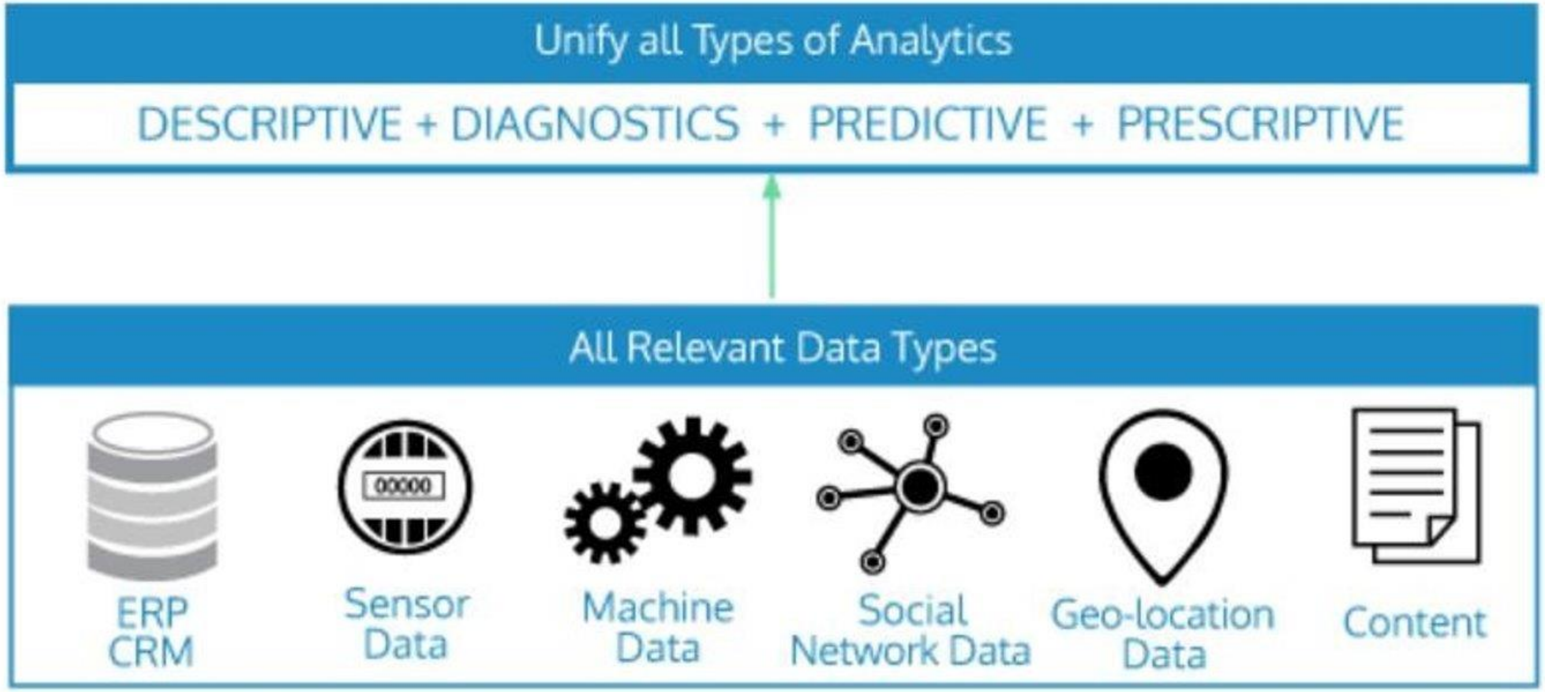
- Real-time visibility of 300,000+ trucks in order to improve uptime and vehicle performance
- OnCommand Connection is collecting telematics and geolocation data across the fleet
- Reduced maintenance costs to $.03 per mile from $.12-$.15 per mile
- Centralizing data from 13 systems with varying frequency and semantic definitions

Data powered CPS Analytics Approach

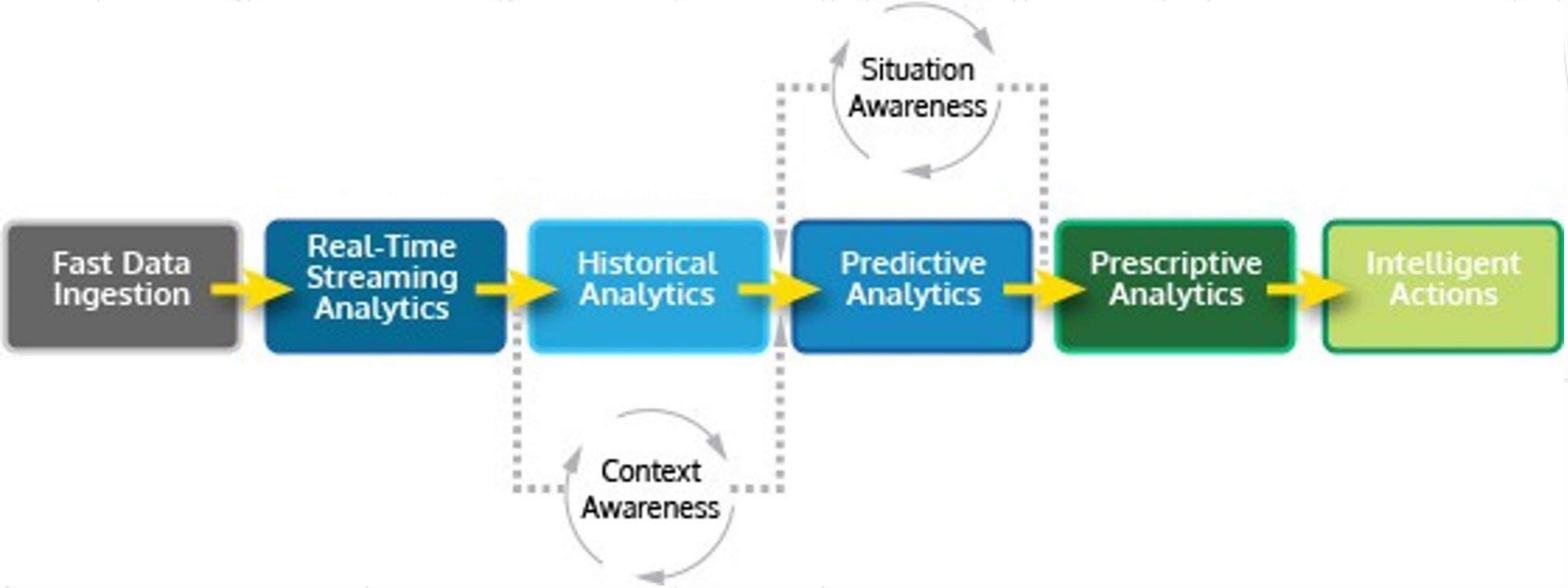
Foundational Steps

1.Simplify the process by integrating all the data for an CPS application

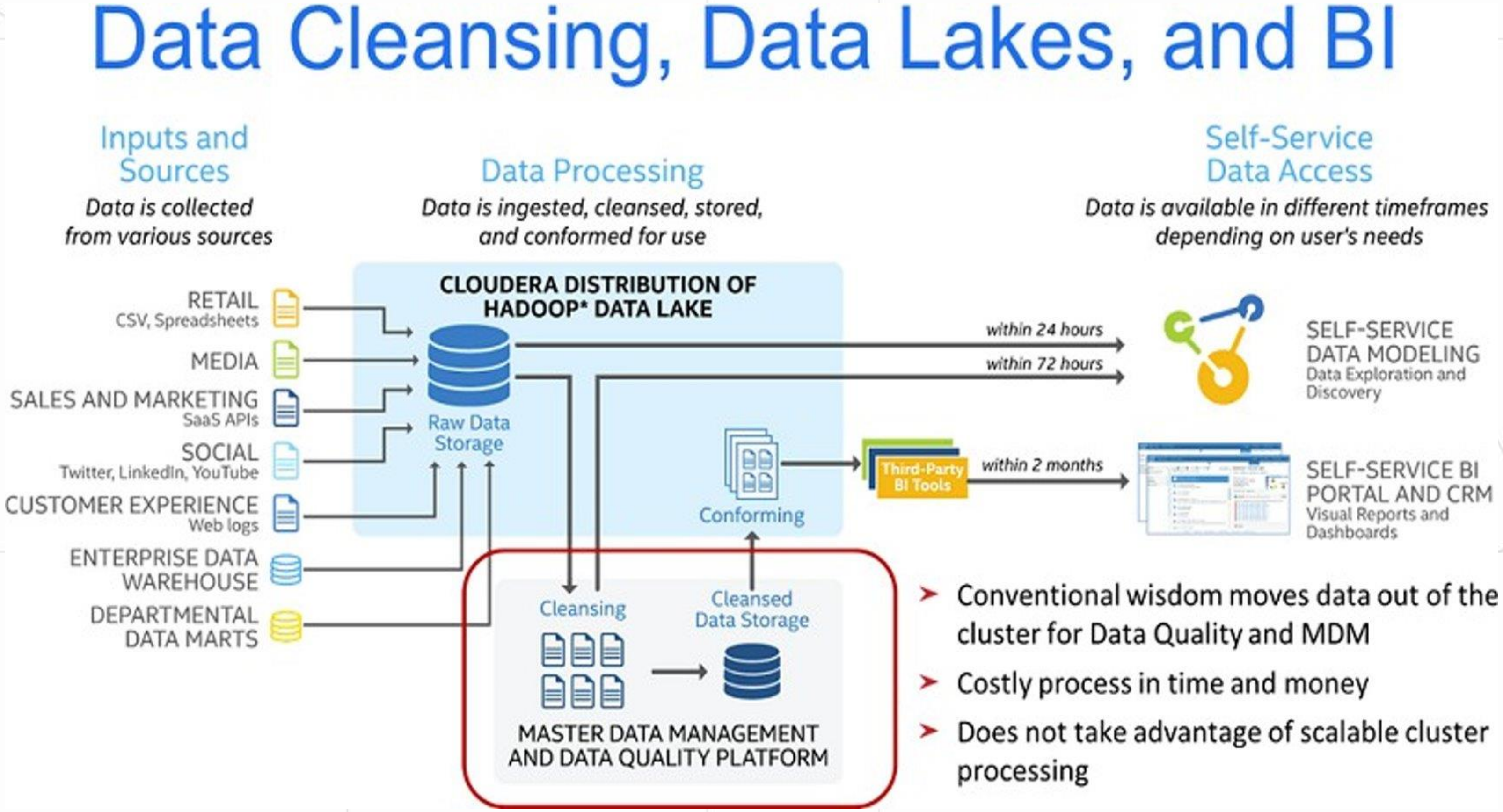
2.Streamline process is to unify the analytics layer



Steps of CPS Analysis



Data Management Platform



Data Cleansing, Data Lakes, and BI

Inputs and Sources
Data is collected from various sources
Data Processing
Data is ingested, cleansed, stored, and conformed for use
Self-Service Data Access
Data is available in different timeframes depending on user's needs
CLOUDERA DISTRIBUTION OF HADOOP* DATA LAKE
RETAIL
CSV, Spreadsheets
MEDIA
SALES AND MARKETING
SaaS APIs
SOCIAL
Twitter, LinkedIn, YouTube
CUSTOMER EXPERIENCE
Web logs
ENTERPRISE DATA WAREHOUSE
DEPARTMENTAL DATA MARTS
Raw Data Storage
Conforming
Cleansing
Cleansed Data Storage
MASTER DATA MANAGEMENT AND DATA QUALITY PLATFORM
within 24 hours
within 72 hours
within 2 months
Third-Party BI Tools
SELF-SERVICE DATA MODELING
Data Exploration and Discovery
SELF-SERVICE BI PORTAL AND CRM
Visual Reports and Dashboards

- Conventional wisdom moves data out of the cluster for Data Quality and MDM
- Costly process in time and money
- Does not take advantage of scalable cluster processing

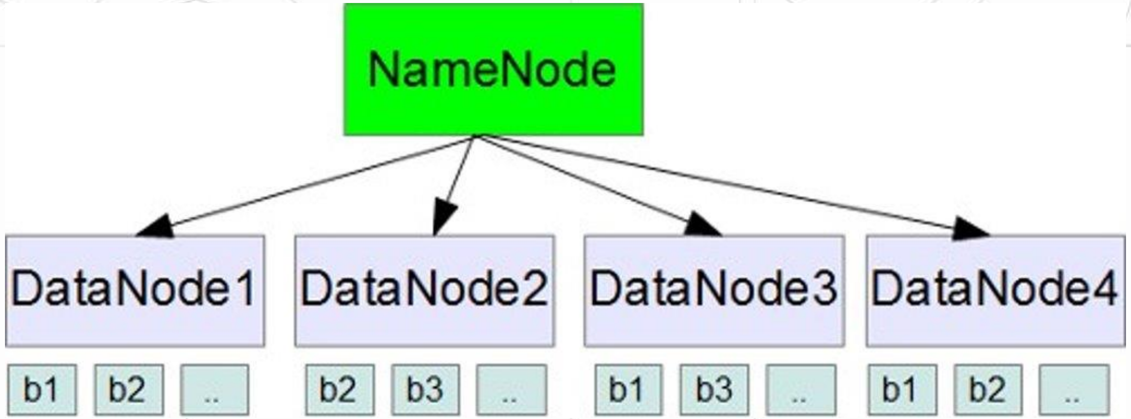
Hadoop Fundamentals

- Hadoop distributed architecture

- Data and processing are distributed across multiple servers
- Each and every server offers local computation and storage. i.e When you run a query against a l arge data set, every server in this distributed architecture will be executing the query on its local machine against the local data set.
- Query is split across multiple servers
- Resultset from all this local servers are consolidated.
- Results of a query on a larger dataset are returned faster.

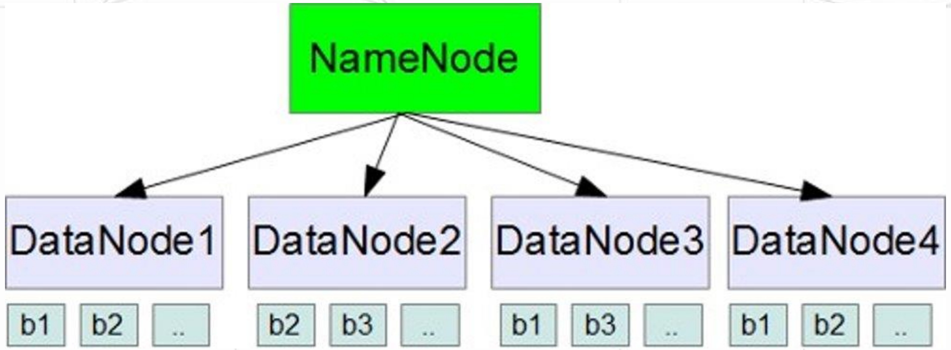
Hadoop Fundamentals

- One NameNode, and multiple DataNodes (servers). b1, b2, indicates data blocks.
- Typical HDFS block size is 128MB.
- When you dump a file (or data) into the HDFS, it stores them in blocks on the various nodes in the Hadoop cluster.
- HDFS creates several replication of the data blocks and distributes them accordingly in the cluster in w ay that will be reliable and can be retrieved faster.



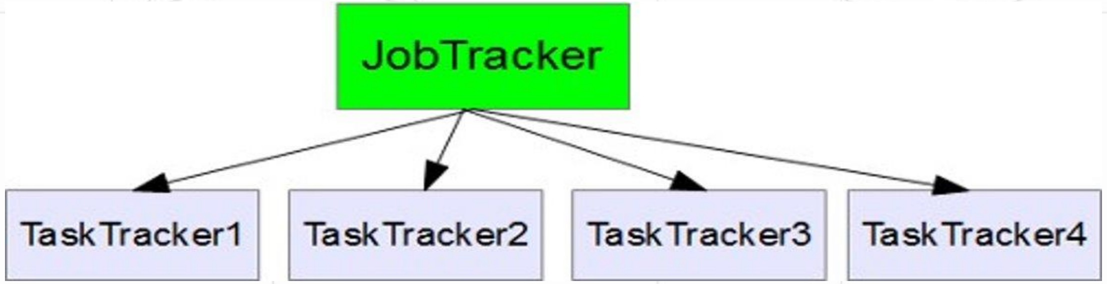
Hadoop Fundamentals

- Each and every data block is replicated to multiple nodes across the cluster.
- Framework make sure that any node failure will never results in a data loss.
- One NameNode that manages the file system metadata
- Multiple DataNodes that will store the data blocks
- When you execute a query from a client, it will reach out to the NameNode to get the file metadata information, and then it will reach out to the DataNodes to get the real data blocks
- Hadoop provides a command line interface for administrators to work on HDFS
- The NameNode comes with an inbuilt web server from where you can browse the HDFS filesystem and view some basic cluster statistics



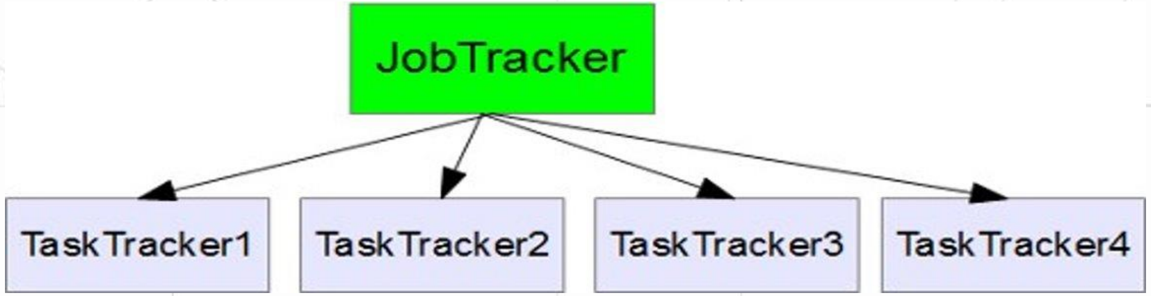
MapReduce

- Parallel programming model that is used to retrieve the data from the Hadoop cluster
- Library handles lot of messy details
  - Parallelization, fault tolerance, data distribution, load balancing, etc.
- Splits the tasks and executes on the various nodes parallel
  - speed up the computation and retrieving required data from a huge dataset in a fast mann er.



MapReduce

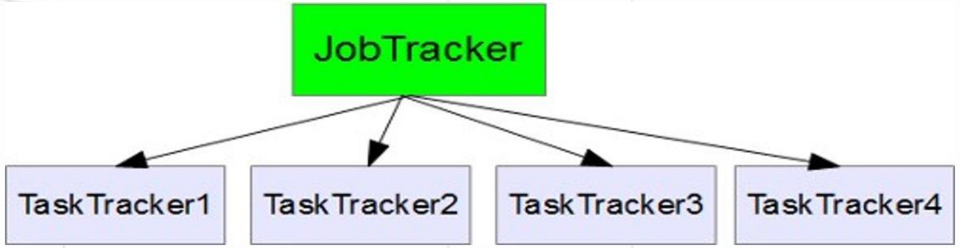
- Developer friendly
  - Implement (or use) two functions: map and reduce
  - In the Mapping step, data is split between parallel processing tasks. Transformation logic can be applied to each chunk of data.
  - Once the mapping is done, all the intermediate results from various nodes are reduced to create the final output



MapReduce

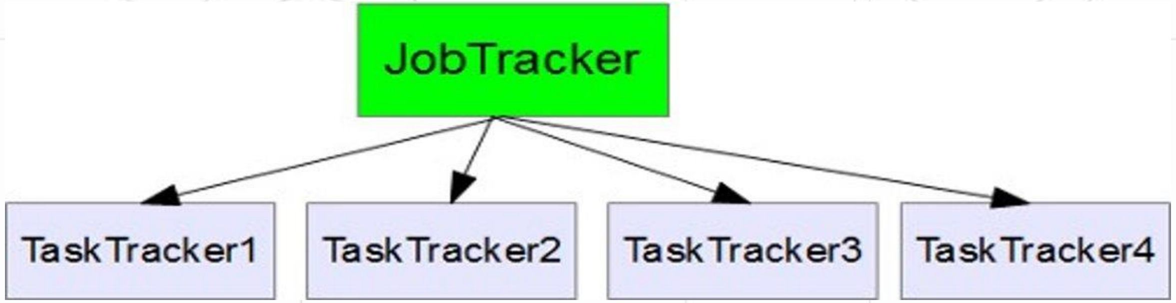
JobTracker

- Keeps track of all the MapReduces jobs that are running on various nodes.
- Schedules the jobs, keeps track of all the map and reduce jobs running across the node s.
- If any one of those jobs fails, it reallocates the job to another node, etc.
- Responsible for making sure that the query on a huge dataset runs successfully and the data is returned to the client in a reliable manner.



MapReduce

- TaskTracker
  - Performs the map and reduce tasks that are assigned by the JobTracker.
  - Constantly sends a heartbeat message to JobTracker
  - Helps JobTracker to decide whether to delegate a new task to this particular node or not.



Apache Spark

- Unified framework to manage big data processing requirements
- Handles variety of data sets that are diverse in nature
  - Text data, graph data etc.
- Handles source of data
  - Batch vs. real-time streaming data
- Enables applications in Hadoop clusters to run
  - Up to 100 times faster in memory
  - 10 times faster even when running on disk

Apache Spark

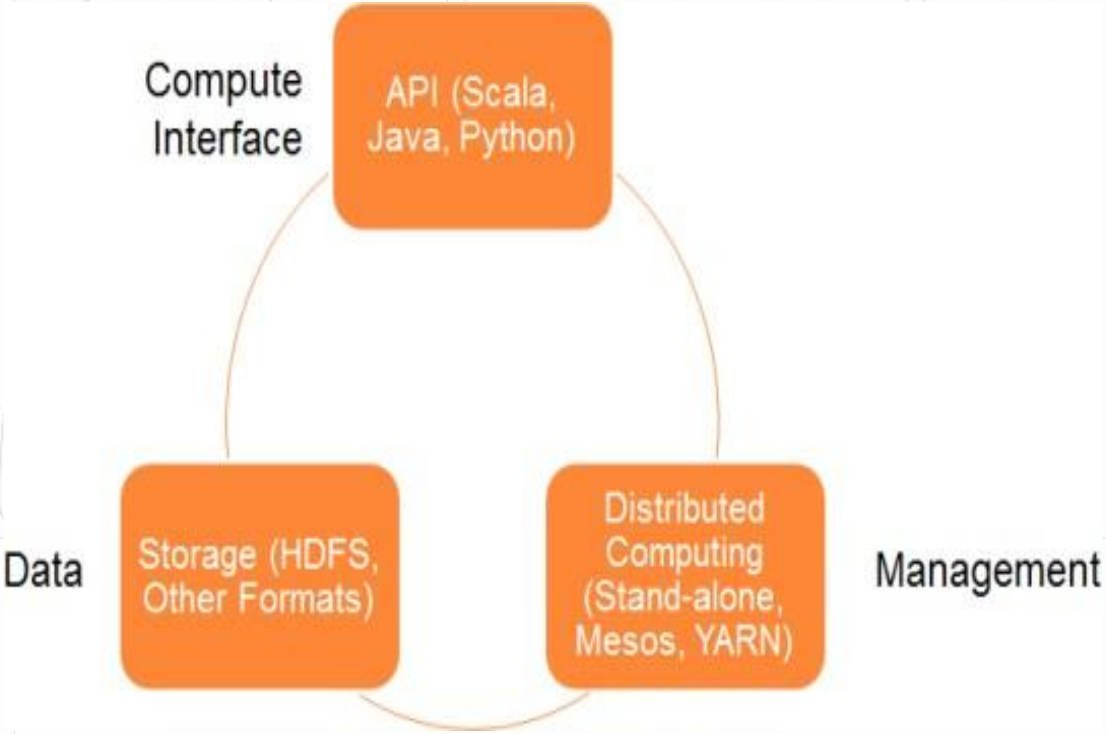
- **Spark Streaming**
  - Processing the real-time streaming data
  - Based on micro batch style of computing and processing
  - Uses the Dstream
    - Series of RDDs, to process the real-time data
- **Spark SQL**
  - Provides the capability to expose the Spark datasets over JDBC API
  - Allow running the SQL like queries on Spark data
    - Using traditional BI and visualization tools
  - Allows users to ETL their data from different formats it’s currently in, transform it, and expose it for ad-hoc querying
    - JSON, Parquet, a Database

Apache Spark

- **Spark Mllib**
  - Scalable machine learning library consisting of common
    - Learning algorithms and utilities, including classification, regression, clustering, collaborative filtering, dimensionality reduction, as well as underlying optimization primitives.
- **Spark GraphX:**
  - Spark API for graphs and graph-parallel computation
  - Support graph computation
  - Collection of graph algorithms and builders to simplify graph analytics tasks

Apache Spark Architecture

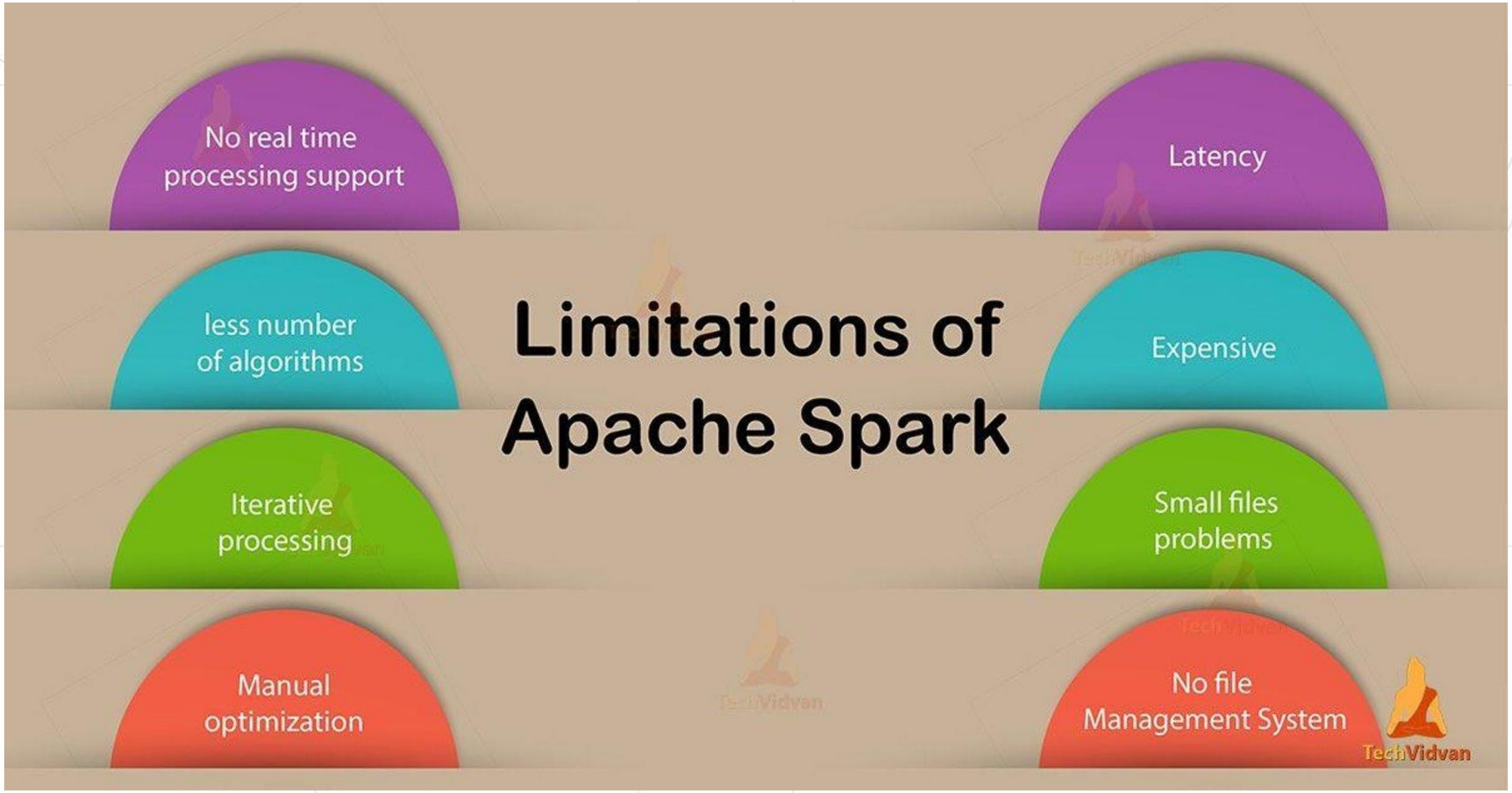
- **Main components**
- Data Storage
- API
- Management Framework


Compute Interface
API (Scala, Java, Python)
Data
Storage (HDFS, Other Formats)
Distributed Computing (Stand-alone, Mesos, YARN)
Management

Apache Spark Core

- **Resilient Distributed Datasets**
- Table in a database
- Hold any type of data
- Stores data in RDD on different partitions
- Help with rearranging the computations and optimizing the data processing
- Fault tolerance
  - Know how to recreate and re-compute the datasets
- Immutable
  - Modify an RDD with a transformation but the transformation returns you a new RDD whereas the original RDD remains the same

Limitations of Spark

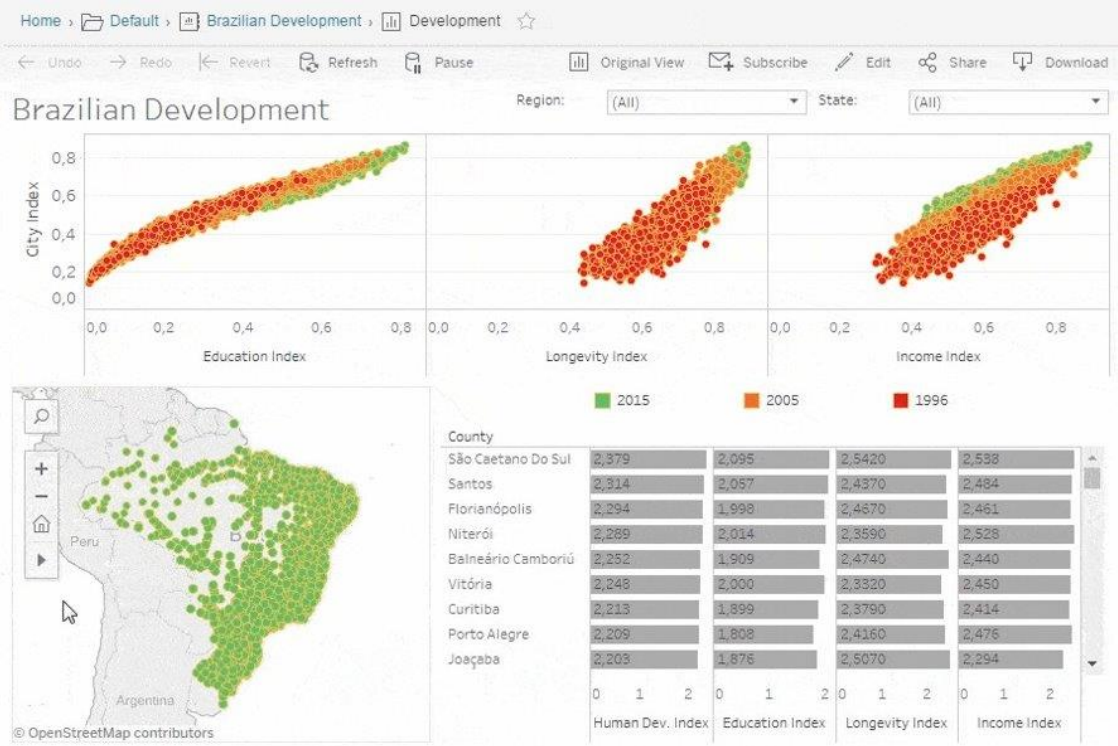


Limitations of Apache Spark

- No real time processing support
- less number of algorithms
- Iterative processing
- Manual optimization
- Latency
- Expensive
- Small files problems
- No file Management System

Data Storage and Visualization

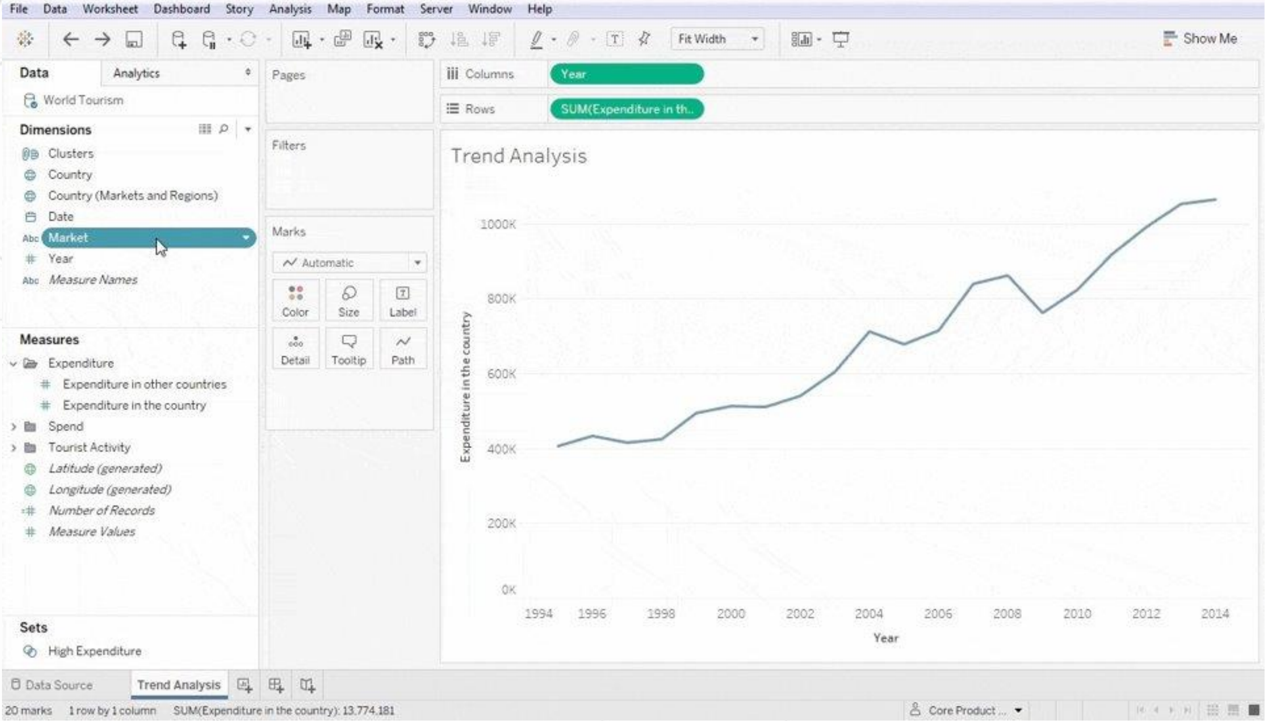
- Processed data can be stored in NoSQL for fast visualization.
- There are several NoSQL databases are available like MongoDB, Cassandra, DynamoDB, Azure DocumentDB, Cloud Bigtable, Cloud Filestore etc.
- For visualization, tools like Tableau, Microsoft Power BI, or Amazon QuickSight can be used.



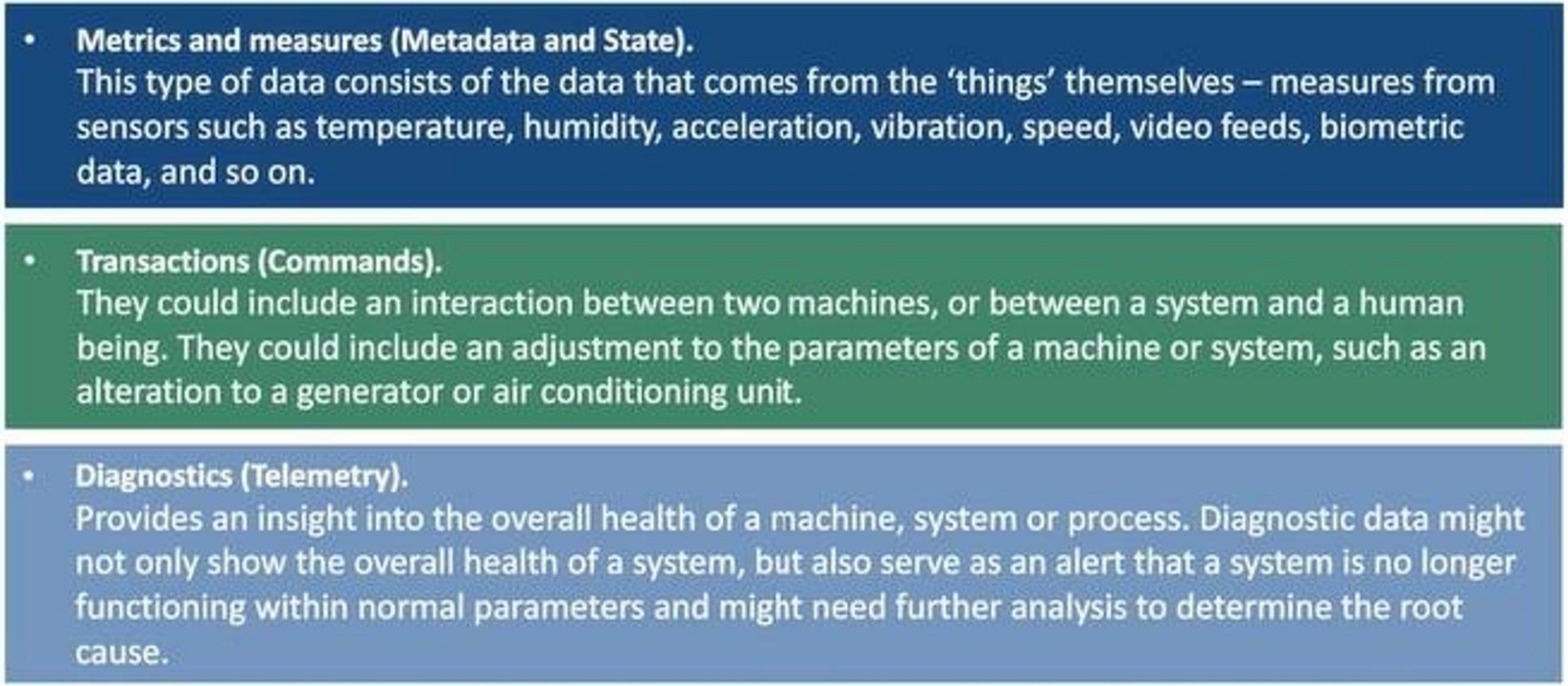
County	Human Dev. Index	Education Index	Longevity Index	Income Index
São Caetano Do Sul	2,379	2,095	2,5420	2,538
Santos	2,314	2,057	2,4370	2,484
Florianópolis	2,294	1,998	2,4670	2,461
Niterói	2,289	2,014	2,3590	2,528
Balneário Camboriú	2,252	1,909	2,4740	2,440
Vitória	2,248	2,000	2,3320	2,450
Curitiba	2,213	1,899	2,3790	2,414
Porto Alegre	2,209	1,808	2,4160	2,476
Joaçaba	2,203	1,876	2,5070	2,294

Data Representation and Visualization

- Data that is generated from heterogeneous systems has heterogeneous visua lization requirements.
- There are currently no satisfactory standard data representation and storage methods that satisfy all of the potential CPS applications



Data of Things



- **Metrics and measures (Metadata and State).**
  This type of data consists of the data that comes from the 'things' themselves – measures from sensors such as temperature, humidity, acceleration, vibration, speed, video feeds, biometric data, and so on.

- **Transactions (Commands).**
  They could include an interaction between two machines, or between a system and a human being. They could include an adjustment to the parameters of a machine or system, such as an alteration to a generator or air conditioning unit.

- **Diagnostics (Telemetry).**
  Provides an insight into the overall health of a machine, system or process. Diagnostic data might not only show the overall health of a system, but also serve as an alert that a system is no longer functioning within normal parameters and might need further analysis to determine the root cause.