Overview of Big Data Technologies

Presented by: WTI
www.wti-solutions.com
703.286.2416

Trusted to Deliver

LEGAL DISCLAIMER

WTI's information publications are made available for educational purposes only as well as to give you general information and a general understanding of technology, not to provide technical advice. By reading our informational publications you understand that there is no contractual relationship created between you and WTI. Although the information in our informational publications is intended to be current and accurate, the information presented herein may not reflect the most current technical developments. These materials may be changed, improved, or updated without notice. WTI is not responsible for any errors or omissions in the content of this publication or for damages arising from the use or performance of this publication under any circumstances. We encourage you to contact us or other technology companies for specific technical advice as to your particular matter.

What is Big Data?

The three Vs of big data- volume, velocity and variety:

- **Volume**: Examples include transaction-based data stored through the years, unstructured data streaming in from social media, collections of sensor and machine-to-machine data. Issues include how to determine relevance within large data volumes and how to use analytics to create value from relevant data.
- **Velocity**: Data streaming in at unprecedented speeds needs to be dealt with in a timely manner and the need to deal with torrents of data in real-time. Examples include RFID tags, sensors and smart metering.
- **Variety:** Data comes in all formats and needs to be managed, merged and governed. e.g. structured data in traditional databases, unstructured text documents, email, video, audio, stock ticker data and financial transactions.

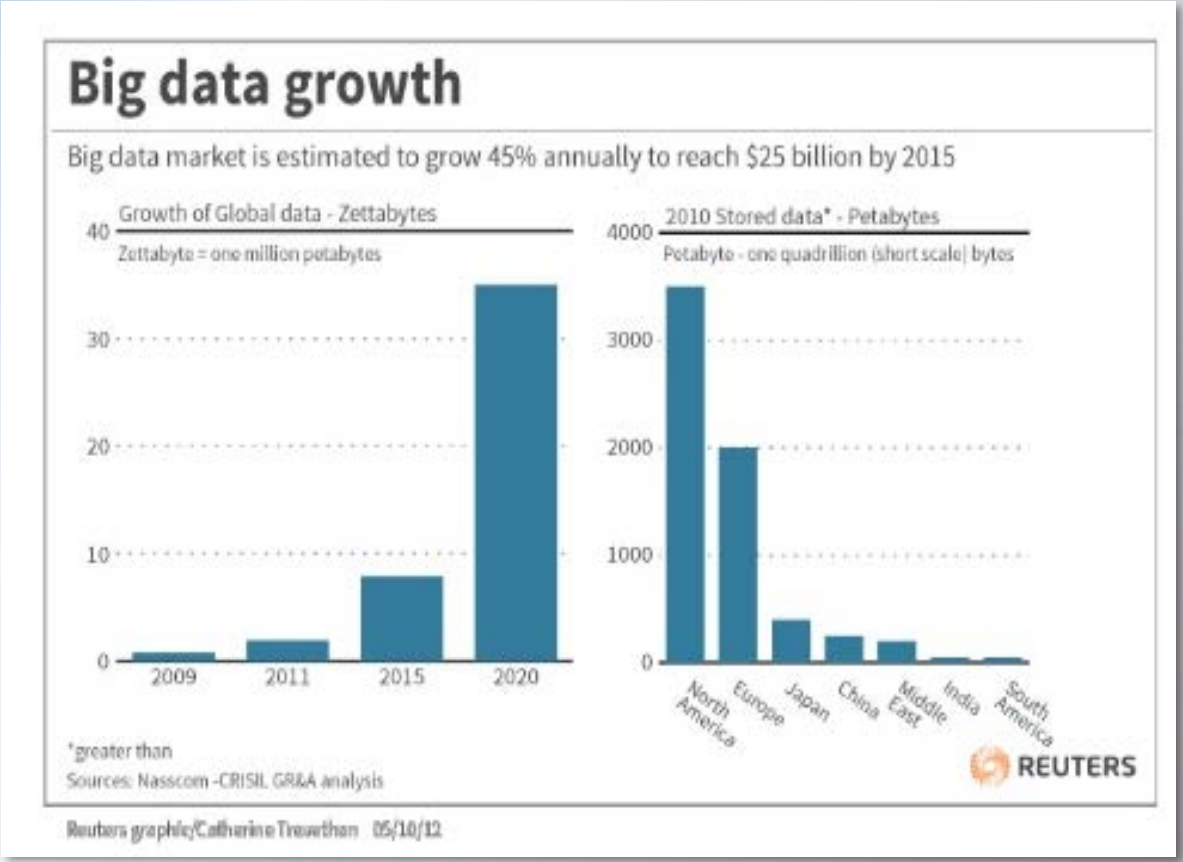
Why Does Big Data Matter?

Organizations will be able to take data from any source, harness relevant data and analyze them to enable:

- **Cost reductions:** saving money to accommodate shrinking budgets.
- **Time reductions:** improving efficiency.
- **New product development and optimized offerings**: developing new products and improving existing solutions.
- **Smarter business decision making:** distilling data for actionable information.

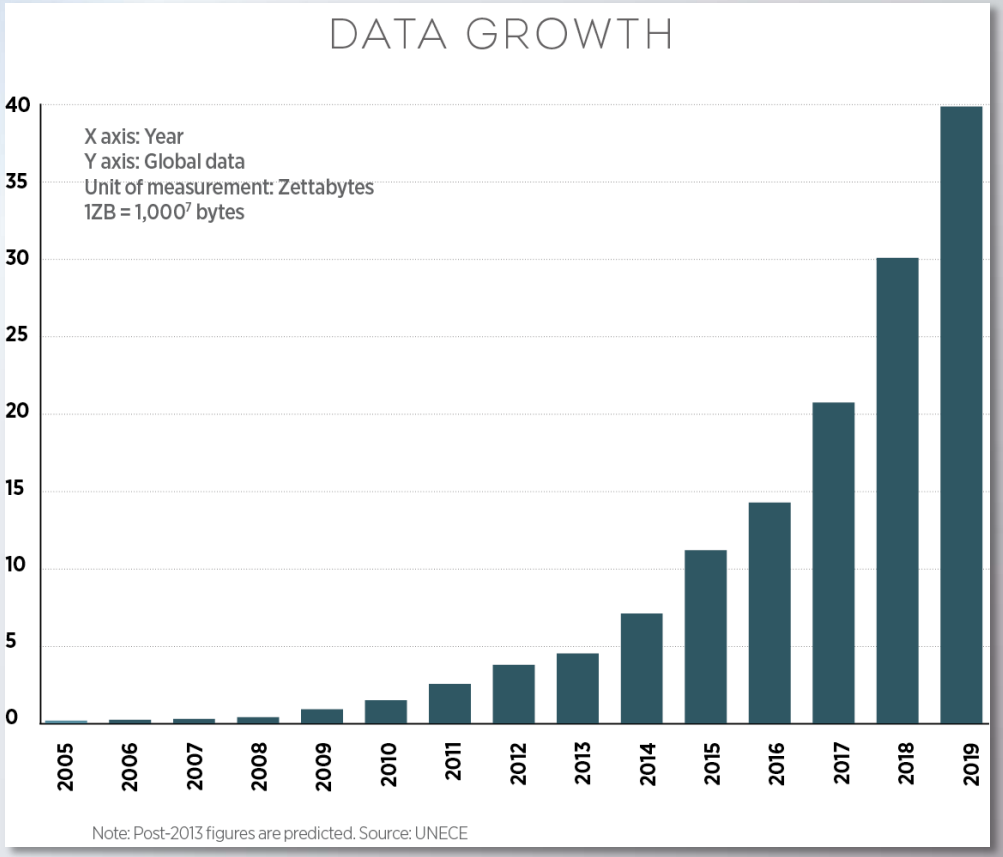
UNPRECEDENTED DATA GROWTH

Due to the continued increase of new data generated and stored, the big data market continues to grow:



UNPRECEDENTED DATA GROWTH

The world's economists predict global data will continue to grow exponentially:



Examples of the Uses of Big Data – Public Sector

Government agencies are using big data technologies to exploit the growth of data and gain new insights. Big data is transforming government in several ways:

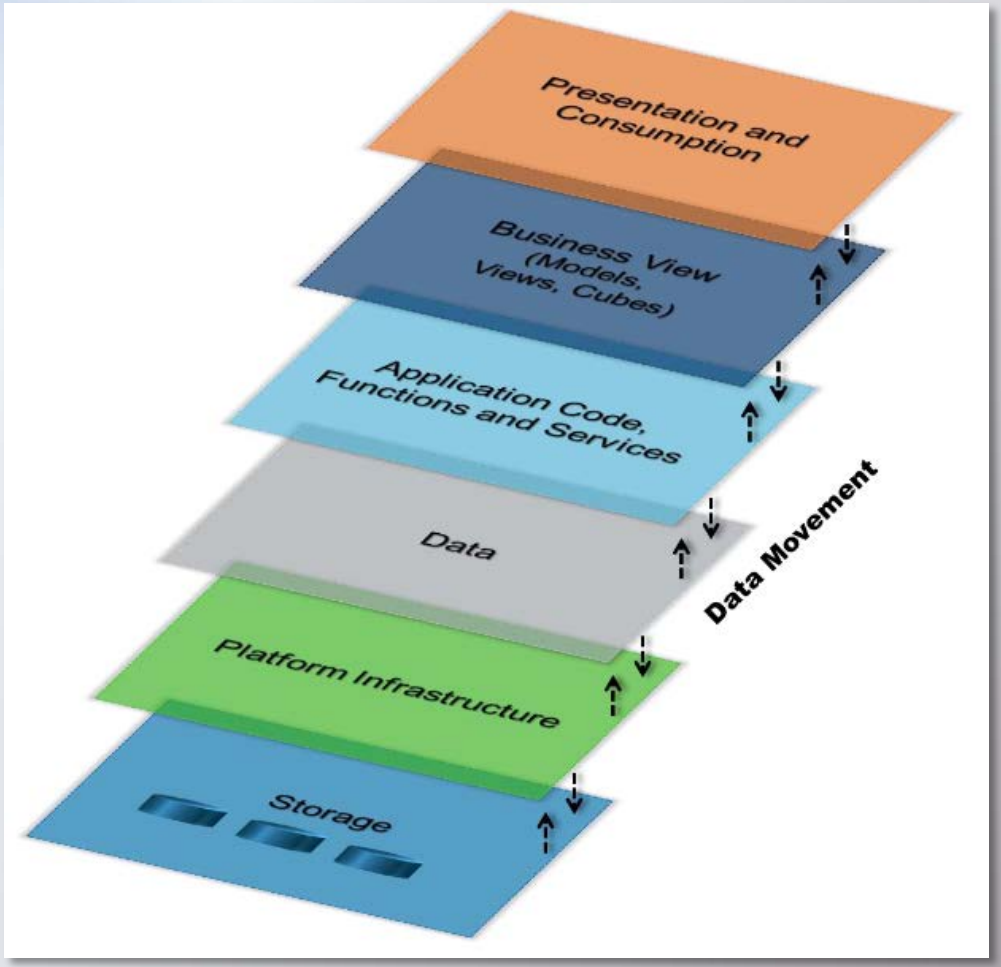
- **Enhancing security and fraud prevention:** Fraud detection can be performed by looking for patterns and anomalies. For example, Customs and Border Protection at the Department of Homeland Security are using big data tools to analyze cargo traffic from entry and exit points to ensure that the global supply chain is secure. DARPA is using big data analytics on cyber data to detect pattern recognition and anomalies to determine threats to computer networks. Using big-data analysis, security experts can track patterns and discover unknown threats in real time, incident response teams can monitor known threats and other suspicious behavior, and law enforcement agents can correlate historical data to help prevent future cases of fraud.
- **Service delivery improvements and emergency response:** Big-data technologies are helping government agencies be more productive, work smarter and be more agile at a time when budgets are tighter than ever. At the state and local level, big data tools are used to monitor complex transportation systems. Real-time analysis allows officials to anticipate problems that could disrupt transportation flow, alleviate traffic congestion and address other transit issues. Big-data technologies are helping cities refine their disaster response and information-collecting mechanisms by improving the way people funnel their communications through 911 and 311 lines.
- **Democratizing information:** The Obama administration recently issued an executive order for open-data. As a result, the health care industry is starting to shift toward more patient-centric systems in which people receive the most timely, appropriate treatment available. There will be more open health care data that empowers patients. People will have access to information about which hospitals provide the most cost-effective procedures with the best outcomes to enable informed decision-making.

Examples of the Uses of Big Data – Private Sector

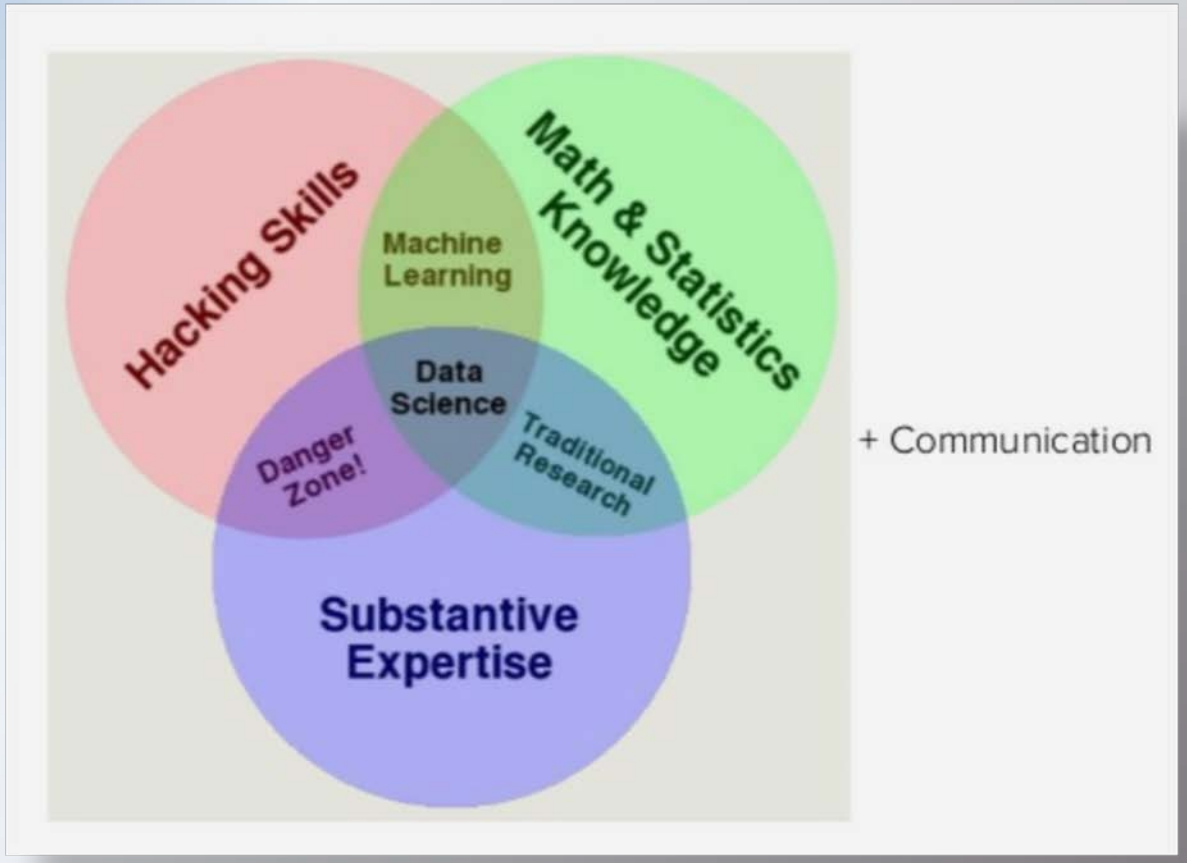
The private sector is using big data technologies to reveal patterns, trends, and associations especially relating to human behavior and interactions – to lower overhead and increase profit in the commercial realm:

- **Determine the root causes of failures**, issues and defects in near-real time, potentially saving billions of dollars annually.
- **Optimize routes for many thousands of package delivery vehicles** while they are on the road.
- **Analyze millions of SKUs to determine prices that maximize profit** and clear inventory.
- **Generate retail coupons at the point of sale** based on the customer's current and past purchases.
- **Send tailored recommendations to mobile devices** while customers are in the right area to take advantage of offers.
- **Recalculate entire risk portfolios** in minutes.
- **Quickly identify customers** who matter the most.
- **Detect fraudulent behavior** via clickstream analysis and data mining.

Big Data Stack

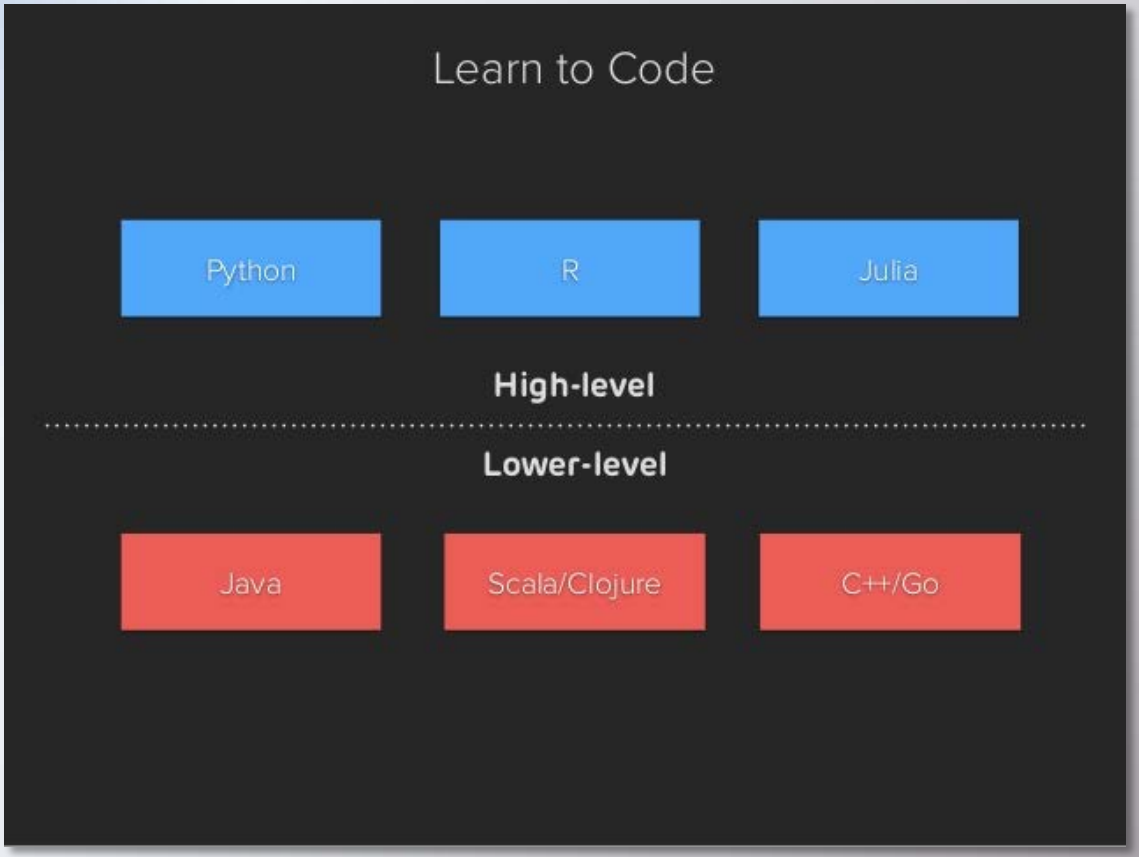


WHAT IS DATA SCIENCE?



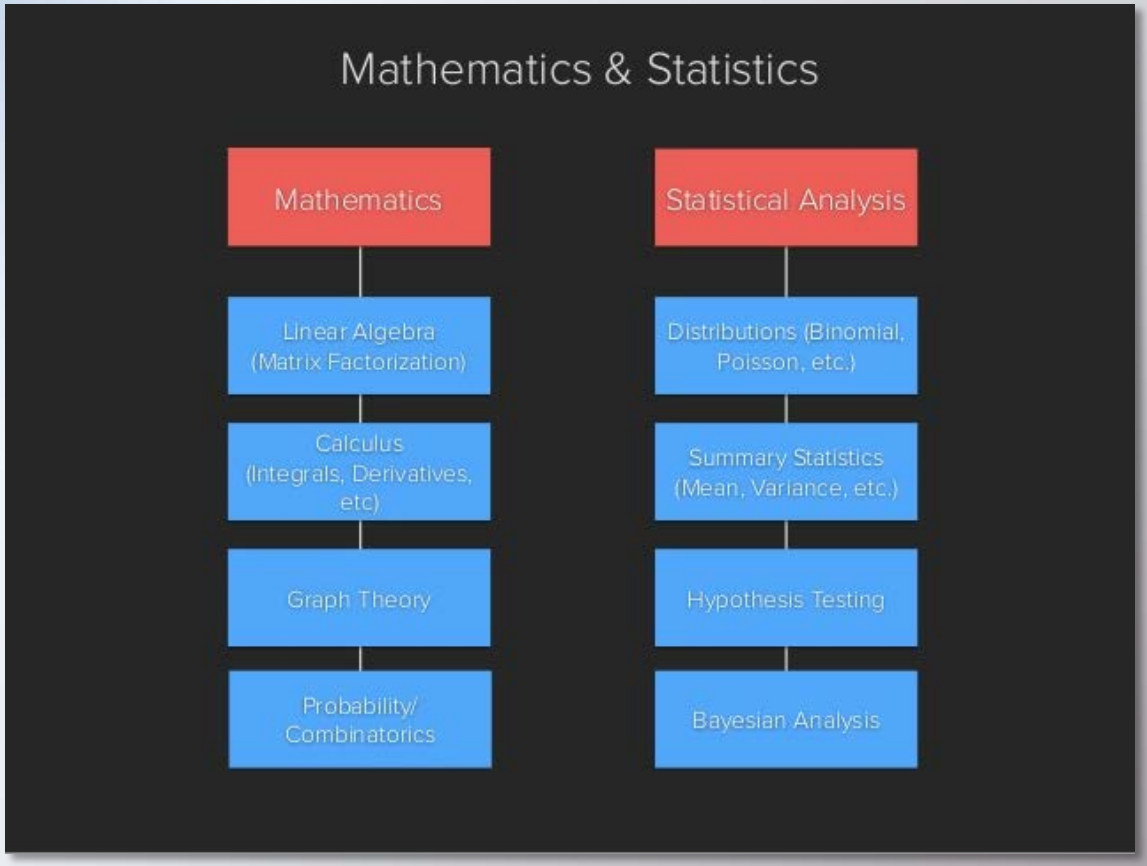
Big Data Coding

These are the most commonly used programming languages and tools by data scientists to perform analytics on large data sets:



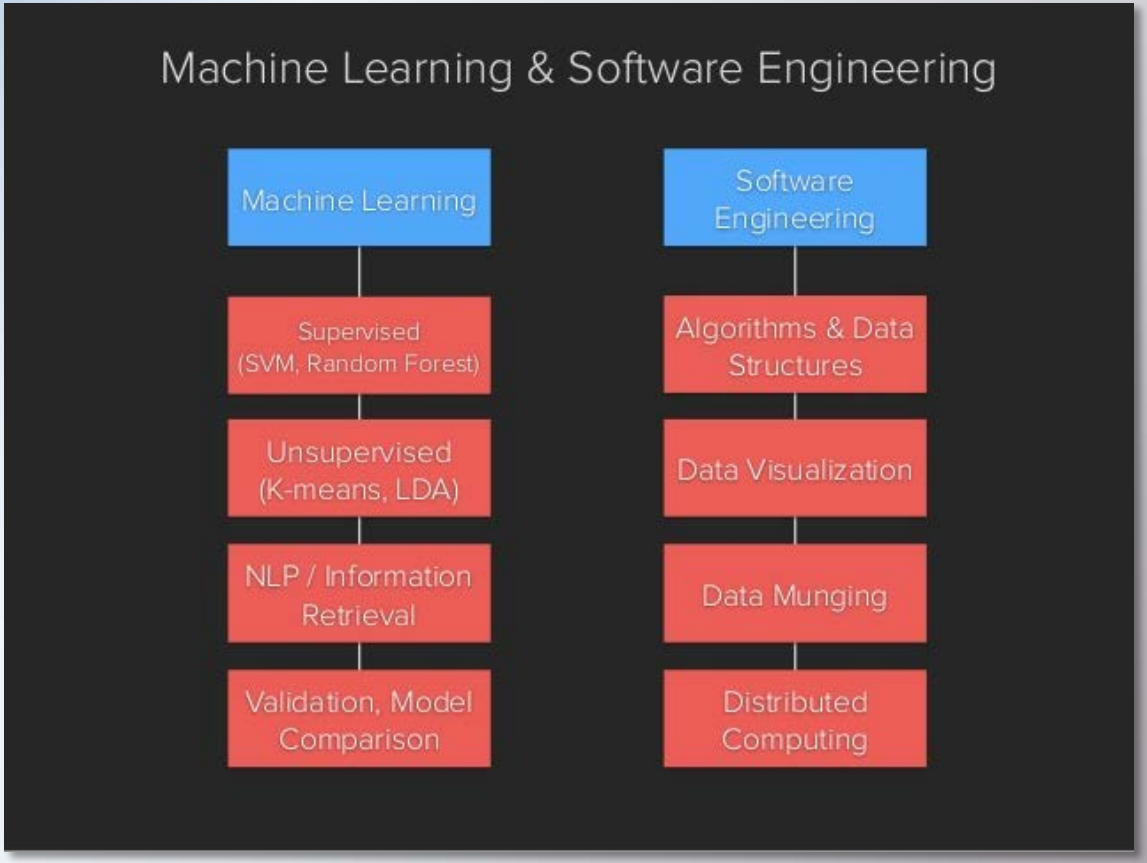
Big Data Mathematics and Statistics

These are the mathematical and statistical techniques that are the theoretical underpinnings of bi data machine learning algorythms:



Machine Learning and Software Engineering

These are the machine learning and software algorythms and techniques used to perform analytics:



What is Hadoop®?

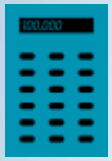
- **Apache™ Hadoop® is an open source software project** that enables distributed processing of large data sets across clusters of commodity servers.
- **It is designed to scale up from a single server to thousands of machines,** with a very high degree of fault tolerance.
- **Rather than relying on high-end hardware,** the resiliency of these clusters comes from the software's ability to detect and handle failures at the application layer.

Why Use Hadoop®?

Hadoop® changes the economics and dynamics of large-scale computing. It's impact can be boiled down to four salient characteristics:

Scalable: A cluster can be expanded by adding new servers or resources without having to move, reformat or change the dependent analytic workflows or applications.

Flexible: Hadoop is schema-less and can absorb any type of data, structured or not, from any number of sources. Data from multiple sources can be joined and aggregated in arbitrary ways enabling deeper analyses than any one system can provide.



Cost-effective: Hadoop brings massively parallel computing to commodity servers. The result is a sizeable decrease in the cost per terabyte of storage, which in turn makes it affordable to model all of your data.

Fault-tolerant: When you lose a node, the system redirects work to another location of the data and continues processing without missing a beat.

Hadoop® Architecture

Hadoop® is composed of four core components—Hadoop Common, Hadoop Distributed File System (HDFS), MapReduce and YARN:

- **Hadoop Common:** A module containing the utilities that support the other Hadoop components.
- **HDFS:** A file system that provides reliable data storage and access across all the nodes in a Hadoop cluster. It links together the file systems on many local nodes to create a single file system.
- **MapReduce:** A framework for writing applications that process large amounts of structured and unstructured data in parallel across a cluster of thousands of machines, in a reliable, fault-tolerant manner.
- **Yet Another Resource Negotiator (YARN):** The next-generation MapReduce, which assigns CPU, memory and storage to applications running on a Hadoop cluster. It enables application frameworks other than MapReduce to run on Hadoop, opening up a wealth of possibilities.

WHAT IS HDFS?

- **HDFA allows to scale a Hadoop® cluster** to hundreds (and even thousands) of nodes.
- **Data in a Hadoop cluster is broken down into smaller pieces (called blocks) and distributed throughout the cluster**. In this way, the map and reduce functions can be executed on smaller subsets of your larger data sets, and this provides the scalability that is needed for big data processing.
- **Goal of Hadoop is to use commonly available servers in a very large cluster, where each server has a set of inexpensive internal disk drives.** For higher performance, MapReduce tries to assign workloads to these servers where the data to be processed is stored. This is known as data locality.

What is MapReduce?

MapReduce is the heart of Hadoop®. It is this programming paradigm that allows for massive scalability across hundreds or thousands of servers in a Hadoop cluster. MapReduce actually refers to two separate and distinct tasks that Hadoop programs perform:

- **The map job** takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs).
- **The reduce job** takes the output from a map as input and combines those data tuples into a smaller set of tuples. As the sequence of the name MapReduce implies, the reduce job is always performed after the map job.

Data Access Projects for Hadoop®

- **Pig:** A programming language designed to handle any type of data, helping users to focus more on analyzing large data sets and less on writing map programs and reduce programs.
- **Hive:** A Hadoop runtime component that allows those fluent with SQL to write Hive Query Language (HQL) statements, which are similar to SQL statements. These are broken down into MapReduce jobs and executed across the cluster.
- **Flume:** A distributed, reliable and available service for efficiently collecting, aggregating and moving large amounts of log data. Its main goal is to deliver data from applications to the HDFS.
- **Spark:** An open-source cluster computing framework with in-memory analytics performance that is up to 100 times faster than MapReduce, depending on the application.
- **Hcatalog:** A table and storage management service for Hadoop data that presents a table abstraction so the user does not need to know where or how the data is stored.
- **Hbase**: A column-oriented non-relational (noSQL) database that runs on top of HDFS and is often used for sparse data sets.
- **Sqoop:** An ELT tool to support the transfer of data between Hadoop and structured data sources.
- **Avro**: An Apache open source project that provides data serialization and data exchange services for Hadoop.
- **Mahout**: a library of machine learning and data mining algorithms used for processing data stored in the HDFS.

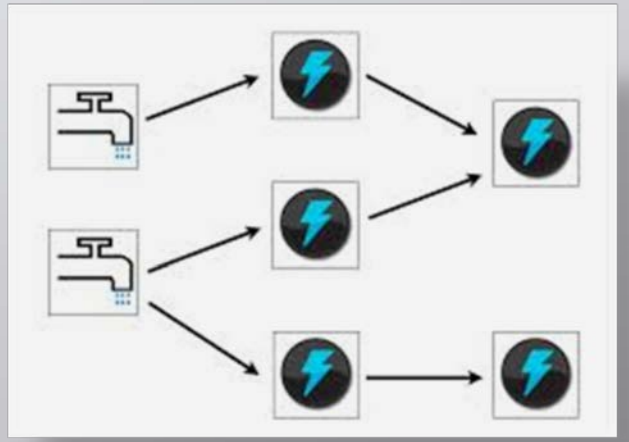
Search Projects for Hadoop®

Solr - An enterprise search tool from the Apache Lucene project that offers powerful search capabilities, including:

- hit highlighting
- indexing capabilities
- reliability
- scalability
- central configuration system
- failover and recovery

HADOOP® ALTERNATIVES: STORM

Example of Storm Topology



- **Also open source** and used by Twitter, Alibaba, Groupon and several other companies.
- **Runs topologies rather than MapReduce jobs.**
  - Topologies run forever until killed by the user.
  - Topology can be visualized as a graph of computation, processing data streams.
  - The sources of these data streams are called "spouts" (symbolized as taps), and they are linked to "bolts" (symbolized by lightning bolts). A bolt consumes any number of input streams, does some processing and potentially emits new streams.
- **Storm topologies are usually programs** written in Java, Ruby, Python and Fancy.
- **Storm software** is written in Java and Clojure.

Other Hadoop® Alternatives – Spark, BashReduce, Disco Project

- **Spark:** Developed by the UC Berkeley AMP lab. Makes data analytics fast in both writing and running. It allows in-memory querying of data instead of just disk I/O and performs better than Hadoop on many iterative algorithms. It is implemented in Scala.
- **BashReduce:** Being just a script, BashReduce implements MapReduce for standard Unix commands (e.g., sort, awk, grep, join, etc.), making it a different alternative to Hadoop. Although it doesn't have a distributed file system at all, BashReduce distributes files to worker machines with an inevitable lack of fault-tolerance.
- **Disco Project:** MapReduce jobs are written in simple Python, while Disco's backend is written in Erlang, a scalable functional language with built-in support for concurrency, fault tolerance and distribution, making it ideal for a MapReduce system. Disco distributes and replicates data but it does not have its own file system.

Other Hadoop® Alternatives (Continued) – GraphLab, HPCC Systems, Sector/Sphere

GraphLab: Developed at Carnegie Mellon, designed for machine learning applications. GraphLab has its own version of the map stage, called the update phase. Unlike MapReduce, the update phase can both read and modify overlapping sets of data. Its graph-based approach makes machine learning on graphs more controllable and improves dynamic iterative algorithms.

HPCC Systems: Uses Enterprise Control Language (ECL) to write parallel-processing workflows, a declarative, data-centric language (similar to SQL, Datalog and Pig). It is written in C++ and it has its own distributed file system.

Sector/Sphere: Developed in C++, this system promises high performance 2-4 times faster than Hadoop. It is composed of two parts: Sector - a scalable and secure distributed file system, and Sphere - a parallel data processing engine that can process Sector data files on the storage nodes with very simple programming interfaces. It has good fault-tolerance, WAN support and is compatible with legacy systems.

HADOOP® COMPLEMENTARY PROJECTS

- **Drill:** This Hadoop add-on focuses on providing an interface for interactive analysis of the datasets stored in the Hadoop cluster. It often makes use of MapReduce to perform batch analysis on big data in Hadoop, and through Dremel it is capable of handling much larger datasets very fast through its ability to scale to a very large number of servers.
- **D3.js:** Short for Data Driven Documents, D3.js is an open source JavaScript library that enables the user to manipulate documents that display big data. This collection of programs can create dynamic graphics using Web technologies (e.g., HTML5, SVG and CSS). Also, it provides many visualization methods such as chord diagrams, bubble charts, dendrograms and node-link trees.
- **Kafka:** A messaging system originally developed at LinkedIn, Kafka serves as the basis for the social medium's activity stream and operational data processing pipeline.
- **Julia:** More of a data analysis tool, Julia is designed to be run in a distributed computing environment such as Hadoop. Similar to Matlab, it is robust, easy to use, and very fast.
- **Impala**: a distributed query execution engine designed to run against data that is stored natively in Apache HDFS and Apache HBase. Developed by Cloudera, it focuses on databases and does not make use of MapReduce at all. This allows it to return results in real-time since it avoids the overhead of MapReduce jobs.

NoSQL Databases

NoSQL databases are becoming popular in Big Data Applications because they can deal with unstructured data and queries are performed more quickly as compared to traditional relational databases:

- **NoSQL is next generation databases that are not relational but horizontally scalable** to hundreds of nodes with high availability and transparent load balancing.
- **Motivations for this approach** include simplicity of design, horizontal scaling, and finer control over availability.
- **The data structures used by NoSQL databases** (e.g. key-value, graph, or document) differ from those used in relational databases, making some operations faster in NoSQL and others faster in relational databases.
- **NoSQL databases are increasingly used in big data** and real-time web applications.
- **NoSQL systems are also called "Not only SQL"** to emphasize that they may also support SQL-like query languages.

Types of NoSQL Databases

There are various approaches to classify NoSQL databases, each with different categories and subcategories.

A few examples in each category are:

- **Column**: Accumulo, Cassandra, Druid, HBase, Vertica
- **Document**: Lotus Notes, Clusterpoint, Apache CouchDB, Couchbase, MarkLogic, MongoDB, OrientDB, Qizx
- **Key-value**: CouchDB, Dynamo, FoundationDB, MemcacheDB, Redis, Riak, FairCom c-treeACE, Aerospike, OrientDB, MUMPS
- **Graph**: Allegro, Neo4J, InfiniteGraph, OrientDB, Virtuoso, Stardog
- **Multi-model**: OrientDB, FoundationDB, ArangoDB, Alchemy Database, CortexDB

Object Oriented Programming (OOP) Languages

These OOP languages are among the most commonly used not only in Big Data analysis but throughout the industry. In fact, the Hadoop platform is Java-based:

- Java
- C++
- Ruby
- JavaScript
- Python
- C#

Functional Languages

Focused on the evaluation of functional expressions rather than the use of variables or the execution of commands in achieving their tasks:

- **Examples include**
  - Scala
  - Erlang
  - Haskell
  - Clojure
  - ML
  - OCaml
  - Clean
- **Advantages are that they are easily scalable and much more error free** since they don't use a global workspace.
- **Disadvantages are that they are somewhat slower** for most data science applications than their OOP counterparts.
- **There can be an overlap between functional languages and traditional OOP languages.** For example, Scala is a functional OOP language.

Data Analysis Software

- **R:** It is open source and is straightforward to write and run programs in, often without the need to include loops. Instead, it makes use of vector operations, which can also extend to matrices. This characteristic is known as vectorization and makes sense for data analysis scripts only.
- **Matlab/Octave:** Although Matlab is proprietary, it has a few open source counterparts, the best of which is Octave. Both Matlab and Octave are great for beginners, have a large variety of applications and employ vectorization, just like R. However, the toolboxes (libraries) of Matlab are somewhat expensive, while Octave doesn't have any at all.
- **Python:** Uses three basic packages for data analysis: numpy (Powerful numerical arrays. A foundational package for the next two packages), scipy (scientific, mathematical, and engineering package) and scikit-learn (Easy to use machine learning library).
- **SPSS:** this is one of the best statistical programs available and is widely used in research. Quite easy to learn, it can do any data analysis, though not as efficiently as R. Also, it is proprietary, just like Matlab. It is preferred by academics and industry professionals alike.
- **Stata:** a good option for a statistical package, Stata is one of the favorite tools of statisticians. Also a proprietary piece of software, it has lost popularity since R became more widespread in the data analysis world.

Visualization Software - Tableau

- This software is proprietary and is somewhat costly.
- It allows for fast data visualization, blending and exporting of plots.
- It is very user-friendly, easy to learn, has abundant material on the web, is fairly small in size (< 100 MB) and its developers are very active in educating users via tutorials and workshops.
- It runs on Windows (any version from XP onwards) and has a two-week trial period. In the industry, Tableau appears to have a leading role compared to other data visualization programs.
- Though more suitable for business intelligence applications, it can be used for all kinds of data visualization tasks, and it allows easy sharing of the visualizations it produces via email or online.
- It also offers interactive mapping and can handle data from different sources simultaneously.

Other Visualization Software

- **Spotfire:** A great product by TIBCO, ideal for visual analytics. It can integrate well with geographic information systems and modeling and analytics software, and it has unlimited scalability. Its price is on the same level as Tableau.
- **Qlikview:** A very good alternative, ideal for data visualization and drilldown tasks. It is very fast and provides excellent interactive visualization and dashboard support. It has a great UI and set of visual controls, and it is excellent in handling large datasets in memory. However, it is limited by the RAM available (scalability issue) and is relatively expensive.
- **Prism:** An intuitive Business Intelligence software that is easy to implement and learn. Focusing primarily on business data, it can create dashboards, scoreboards, query reports, etc., apart from the usual types of plots.
- **inZite:** An interesting alternative, offering both appealing visualization and dashboards features. Very fast and intuitive.
- **Birst:** A good option, offering a large collection of interactive visualizations and analytics tools. It can create pivot tables and drill into data with sophisticated, straightforward reporting functions.
- **SAP Business Objects:** This software offers point-and-click data visualization functionality in order to create interactive and shareable visualizations as well as interactive dashboards. Naturally, it integrates directly with other SAP enterprise products.

SUMMARY

- As a technology-neutral and unbiased big data solutions provider with across-the-board expertise and real-world experience, WTI understands how to solve the big data challenges facing the public and private sector.

- WTI has hands-on experience in the field implementing myriad big data technologies as well as expertise in the theoretical underpinnings behind these technologies.

- Contact us to find out how we can help you harness big data into actionable information.

CONTACT INFORMATION

All inquiries regarding this informational publication should be directed to the following:

Cindy Ford
President
703.286.2416 (office)
703.638.2498 (mobile)
703.997.7227 (fax)
cford@wti-solutions.com

Debra O'Connell
Director of Strategic Relationships
703.286.2416 (office)
301.312.1263 (mobile)
703.997.7227 (fax)
debra.oconnell@wti-solutions.com