# Data Science Starter Program
# Introduction to Data Science

E. Le Pennec, A. Fermin


Spring 2015

# Introduction to Data Science

## Outline

# Data Science in the media

## Le Monde

INTERNATIONAL | POLITIQUE | SOCIÉTÉ | ÉCONOMIE | CULTURE | IDÉES | SPORT | SCIENCES | TECHNO | STYLE | VOUS | ÉDITION ABONNÉS

M Idées

IDÉES | Les débats | Think tanks | Points de vue | Editoriaux | Opinions du Monde | Analyses | Idées chroniques | Chats | Blogs | Forums

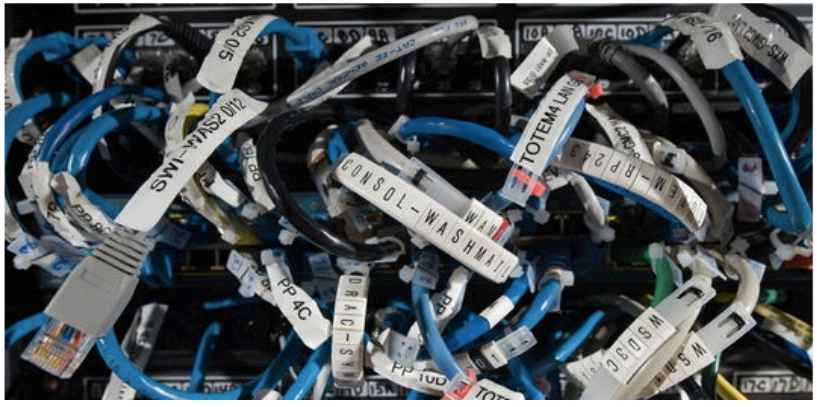
### Les données, puissance du futur

LE MONDE | 07.01.2013 à 15h10 • Mis à jour le 07.01.2013 à 18h03

Par Stéphane Grumbach, Stéphane Frénot

Abonnez-vous à partir de 1 € | Réagir | Classer | Imprimer | Envoyer | Partager

Recommander | Envoyer | 467 personnes le recommandent.




**Les plus partagés**

1. Une équipe de scientifiques filme un calamar géant par 900 mètres de fond dans le Pacifique — 2212
2. Infirmiers et aides-soignants refusent d'être des "pigeons" — 1664
3. Messi remporte son 4e Ballon d'or consécutif — 987
4. Mariage homosexuel : Wauquiez veut "forcer" le débat sur un référendum — 629
5. La première Eglise athéiste ouvre à Londres — 603

**Nous suivre**

Retrouvez le meilleur de notre communauté

# Data Science in the media

## NY Times



OCTOBER 24, 2012, 9:00 AM | 4 Comments

# Big Data in More Hands

By QUENTIN HARDY



Business people, Big Data is coming for you.

Software that captures lots of data and uses it to make predictions has mostly been the province of engineers skilled in arcane databases and statisticians capable of developing complex algorithms. As the business gets bigger, however, software makers are domesticating their products in the hope they will prove attractive to a broader population.

Cloudera, which offers a popular version of the open source database called Hadoop, released software on Wednesday that makes it possible to run queries from a more mainstream SQL programming language interface. SQL, thanks to its adoption by Oracle, Microsoft and others, is known to millions of business analysts.

"This enables us to talk to a whole other class of customer," said Mike Olson, the chief executive of Cloudera. "The knock against Hadoop was that it is too complex."

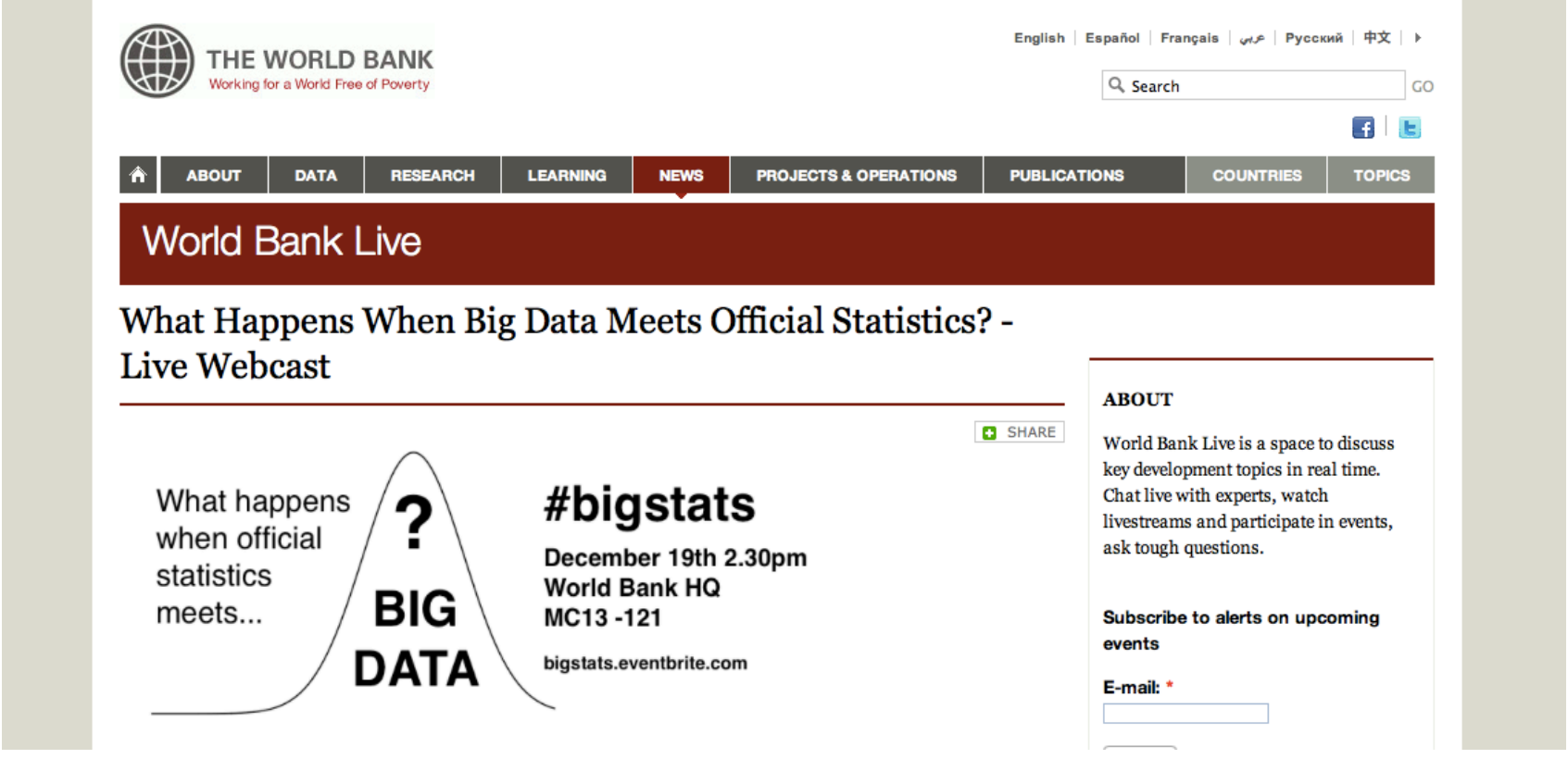
There is a reason for that. Hadoop is one of several so-called unstructured databases that were created at Yahoo and Google, after those two companies found they had previously unimaginable amounts of data about activities like people's Web-surfing habits. Put into databases designed to handle this unstructured behavior, then analyzed, this information was

PREVIOUS POST
Google Shifts Pitch for Its New Chromebooks

NEXT POST
In Contest for Rescue Robots, Darpa Offers $2 Million Prize

# Data Science in the media

## World Bank



# Data Science in the media

## Criteo

criteo

Solutions CPOP Plateforme Editeurs Success Stories Actualités Carrières Contact Login annonceur | FR

Accueil > Actualités & Évènements

### CriteoLabs : soirée d'inauguration

## Criteo inaugure à Paris l'un des premiers centres de R&D en publicité prédictive d'Europe

- Fleur Pellerin, Ministre déléguée chargée des PME, de l'innovation et de l'Economie Numérique, apporte son soutien à cette entreprise innovante du secteur numérique, véritable « success story » à la française.
- Criteo inaugure CriteoLabs, son nouveau centre de R&D de 10.000 m2 au cœur de Paris.
- Avec à terme 300 ingénieurs, ce site est déjà l'un des premiers centres européens de R&D en algorithmes appliqués à la publicité en ligne. Pour accompagner sa forte croissance, Criteo recrute cette année 250 nouveaux collaborateurs.


Criteo inaugure à Paris l'un des plus gros pôles européens de R&D dédiés à la publicité prédictive, CriteoLabs. Sur 10.000 m2, ce nouveau centre a vocation à accueillir 300 ingénieurs et à permettre ainsi à Criteo de garantir son avancée technologique sur ses 30 marchés d'exportation, des Etats-Unis, à l'Europe, en passant par l'Asie. Cette année, l'entreprise compte ainsi recruter 250 nouveaux collaborateurs, dont une centaine d'ingénieurs.

Ce nouveau siège, que Criteo a choisi délibérément de situer à Paris, vient ponctuer un développement continu, qui a permis à l'entreprise d'atteindre des résultats remarquables, 3 ans seulement après son lancement commercial :

- 600 salariés présents dans 15 bureaux dans le monde
- 2 000 annonceurs, parmi les plus importants e-commerçants mondiaux tels que Dell, Macy's, John Lewis, Marks & Spencers, Zalando, La Redoute, Les 3 Suisses, etc.
- 4 000 éditeurs
- Plus de 200 millions de dollars de CA en 2011

Jean-Baptiste Rudelle, CEO et Pascal Gauthier COO

Arrivée de Fleur Pellerin

# Data Science in the media

## We are in the press as well...

L'USINE digitale
Quand le numérique réinvente l'industrie

TOUTE L'INFO | L'USINE NOUVELLE | INSCRIVEZ-VOUS À LA NEWSLETTER | DIGITAL AVENUE Trouvez vos futurs partenaires e-business | Rechercher dans L'Usine Digitale | S'abonner | Suivre

INTERNET | LOGICIELS & APPLICATIONS | HARDWARE | CLOUD ET DATA | INDUSTRIES | ECONOMIE NUMÉRIQUE | ANNUAIRE DE START-UP

USINE DIGITALE > L'USINE CAMPUS

### Polytechnique forme les professionnels au big data

Par Cécile Maillard - Publié le 05 juin 2014, à 15h49

L'Usine Campus, Numérique, Formation, France, L'actu des campus,


Polytechnique forme les professionnels au big data © Wikimedias Commons

**L'Ecole polytechnique ouvre à la rentrée une formation courte pour les doctorants et salariés qui ont besoin de comprendre quelles belles opportunités leur ouvrent les big data. Les écoles d'ingénieurs et de management sont de plus en plus nombreuses à proposer des formations d'analyse des données massives.**

*"Nous avons une pression des entreprises, qui nous questionnent de plus en plus sur ce que propose l'Ecole polytechnique sur les big data"*, explique Frank Pacard, directeur de l'enseignement et de la recherche à l'X. Pour répondre à leur demande, Polytechnique ouvre, en octobre 2014, un parcours baptisé "Data Science Starter Program"*. *"C'est un programme d'évangélisation, qui expliquera à des doctorants, post-doctorants ou salariés, tout ce qu'ils peuvent tirer de l'exploitation des données, dans leurs métiers et entreprises."*

**A LIRE SUR LE MÊME SUJET**

Les quatre conseils d'IBM et SAP pour éviter les pièges du big data

RECEVOIR NOTRE NEWSLETTER :

E-Mail | OK

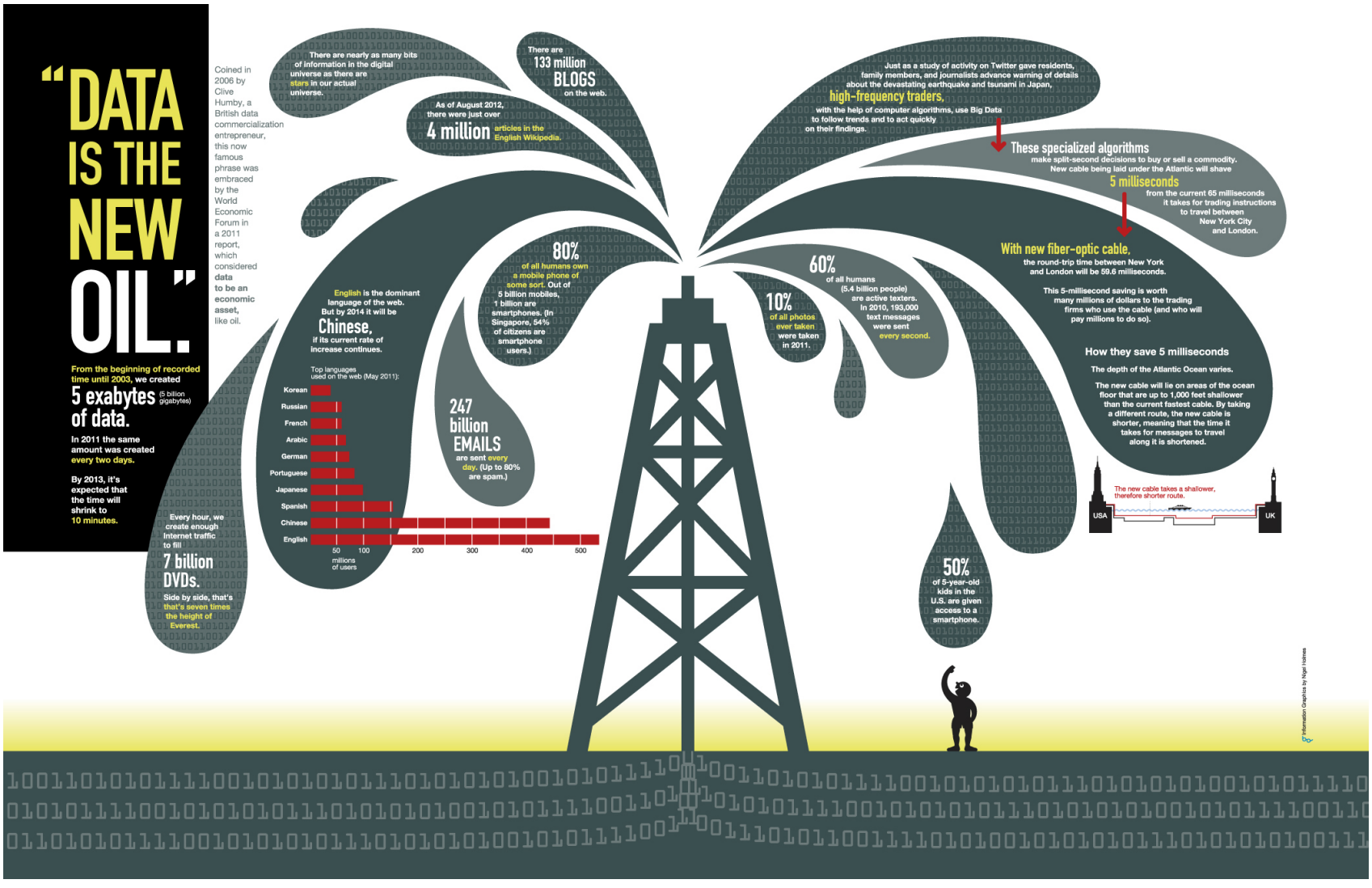
**LES PLUS LUS DE LA RUBRIQUE «NUMÉRIQUE»**

- La SNCF roule au digital
- L'accéléromètre, ce mouchard caché dans nos smartphones
- Le marché européen pourrait faire gagner 18 millions d'abonnés à Netflix d'ici à 2018
- La dataviz, facteur de compétitivité pour les entreprises
- Nous allons révéler la face numérique de l'industrie

**LES AUTRES ACTUALITÉS DE LA RUBRIQUE «NUMÉRIQUE»**

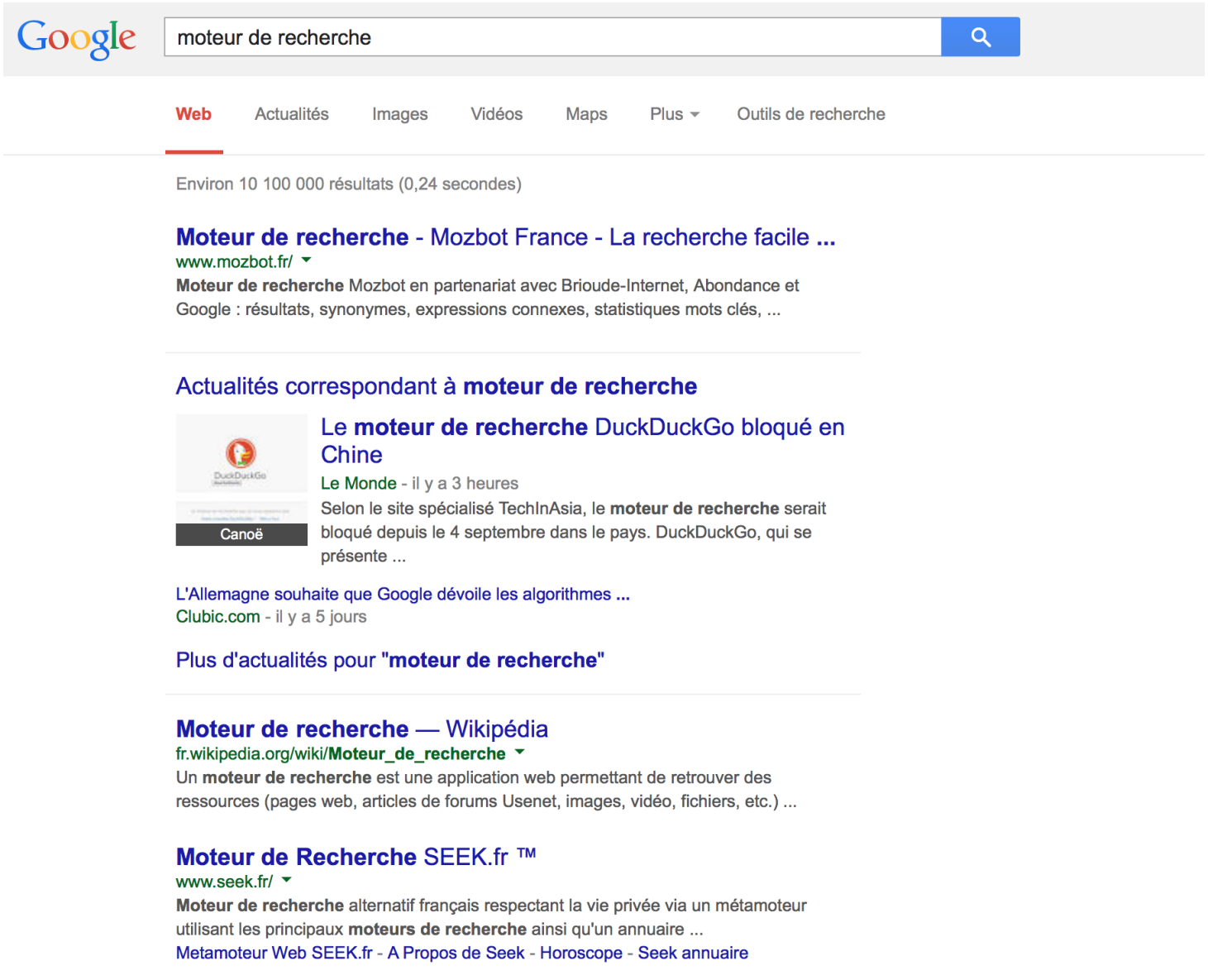
# Data Science in the media

## Data is the new oil?



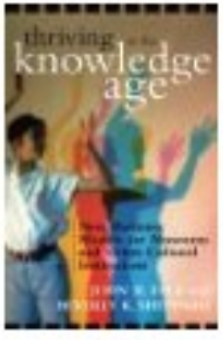
# From Data to Product

## Web search



## More Ideas Based on Your Browsing History

You looked at




Thriving in the Knowledge Age: New... Paperback by John H. Falk
**$29.95**

› Find similar items

You might also consider


Museum Administration: An Introduction Paperback by Hugh H. Genoways
~~$31.95~~ **$28.75**


Exhibit Labels: An Interpretive Approach Paperback by Beverly Serrell
~~$34.95~~ **$27.85**

Recommendations don't have to be about showing you more of the same...

# From Data to Product

## Advertisement





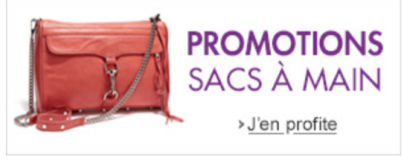












# From Data to Product

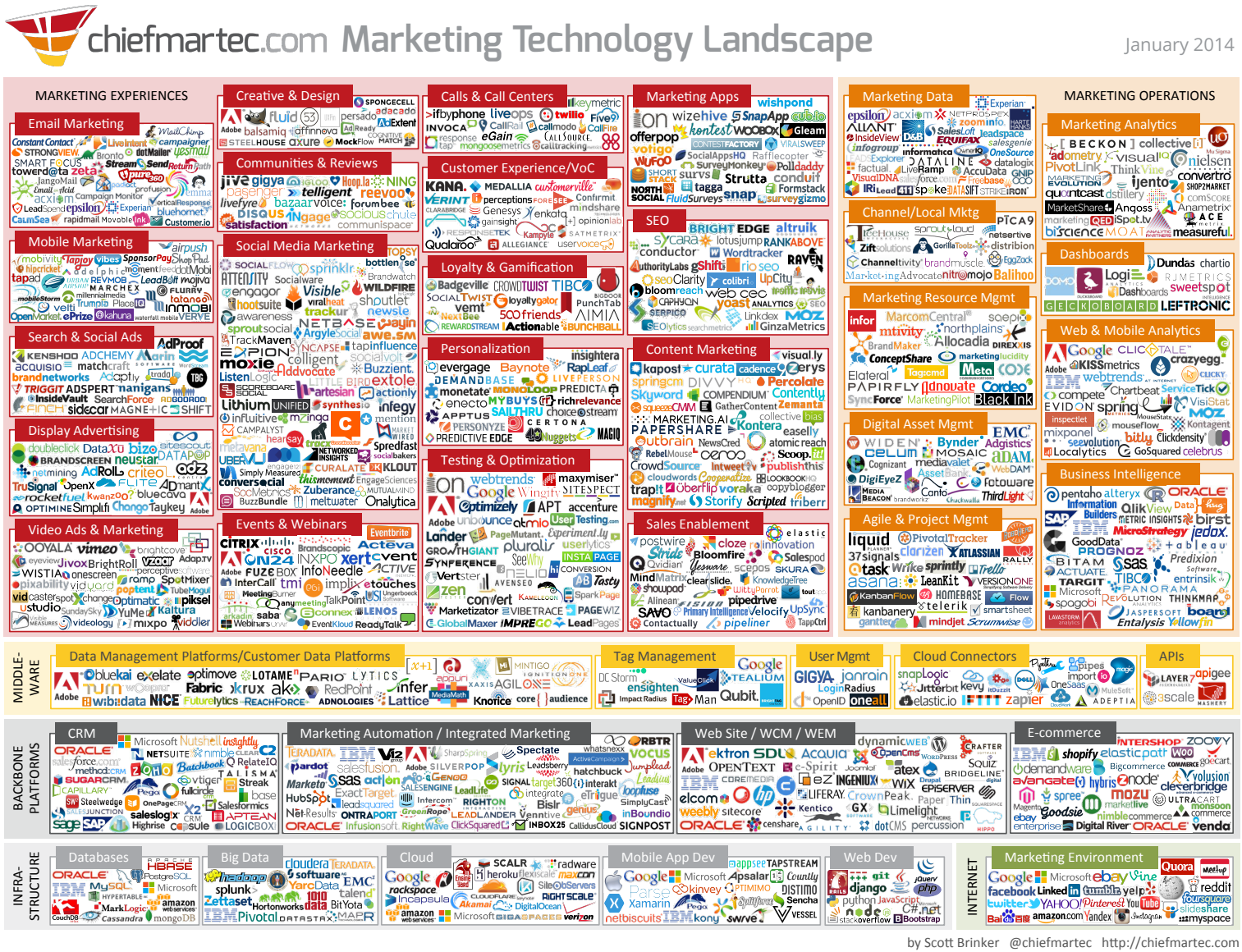
## Intrusion detection

# From Data to Product

## Crime prevention

# From Data to Product

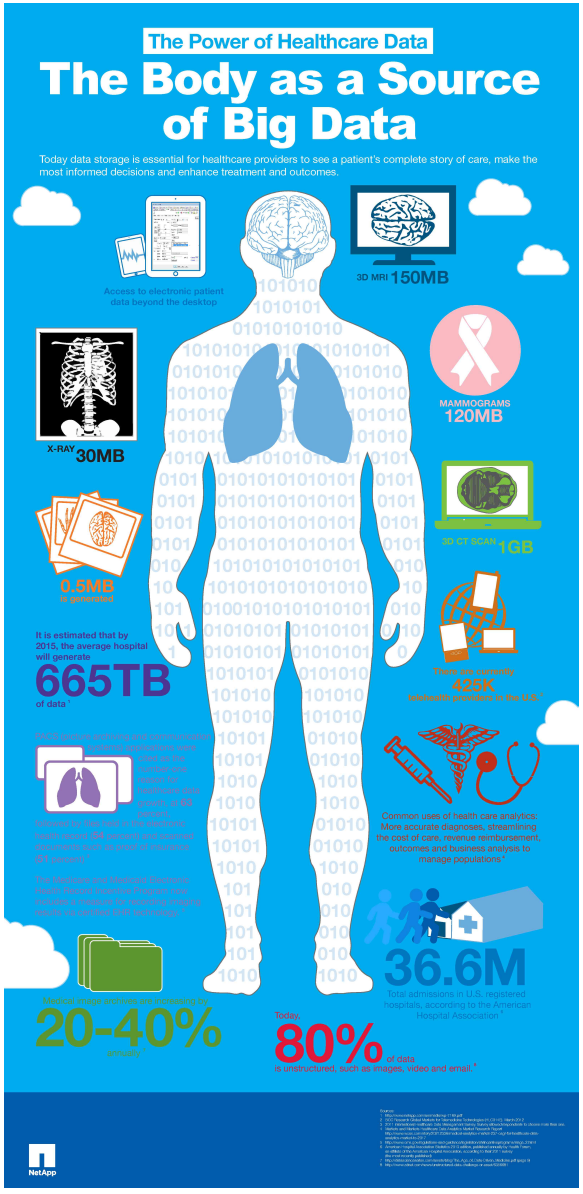
## Marketing




by Scott Brinker @chiefmartec http://chiefmartec.com

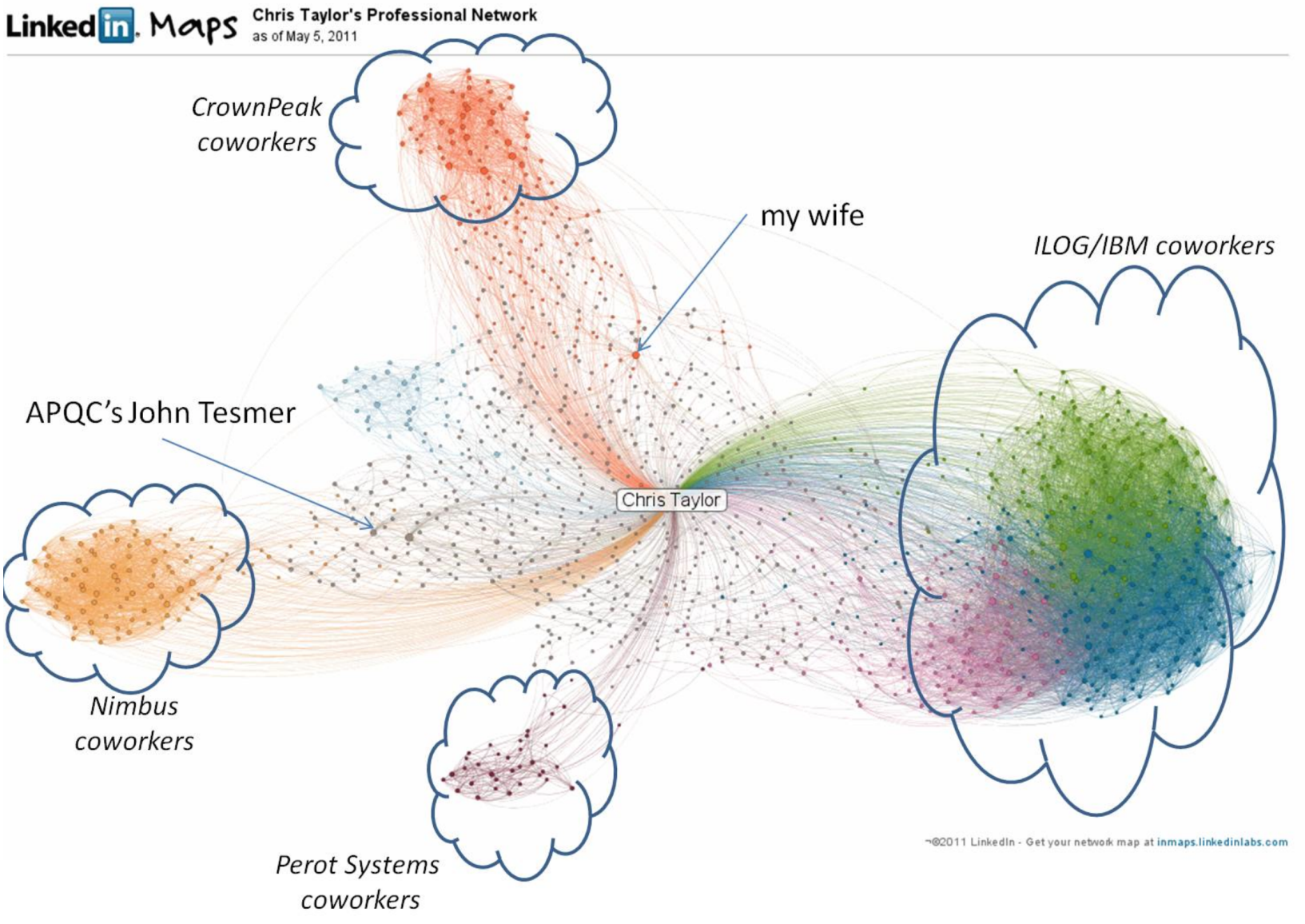
# From Data to Product

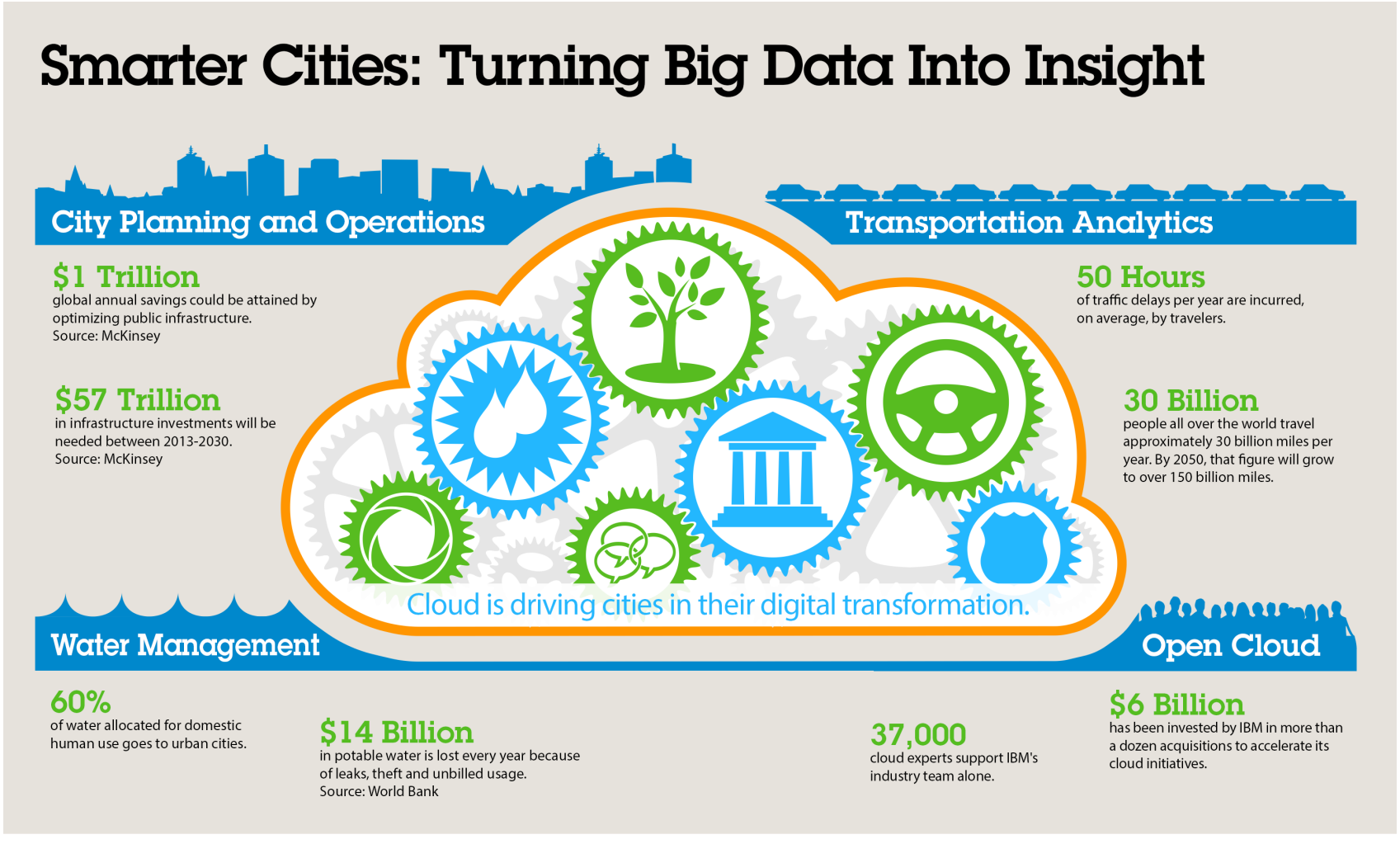
## Health



# From Data to Product

## Linkedin

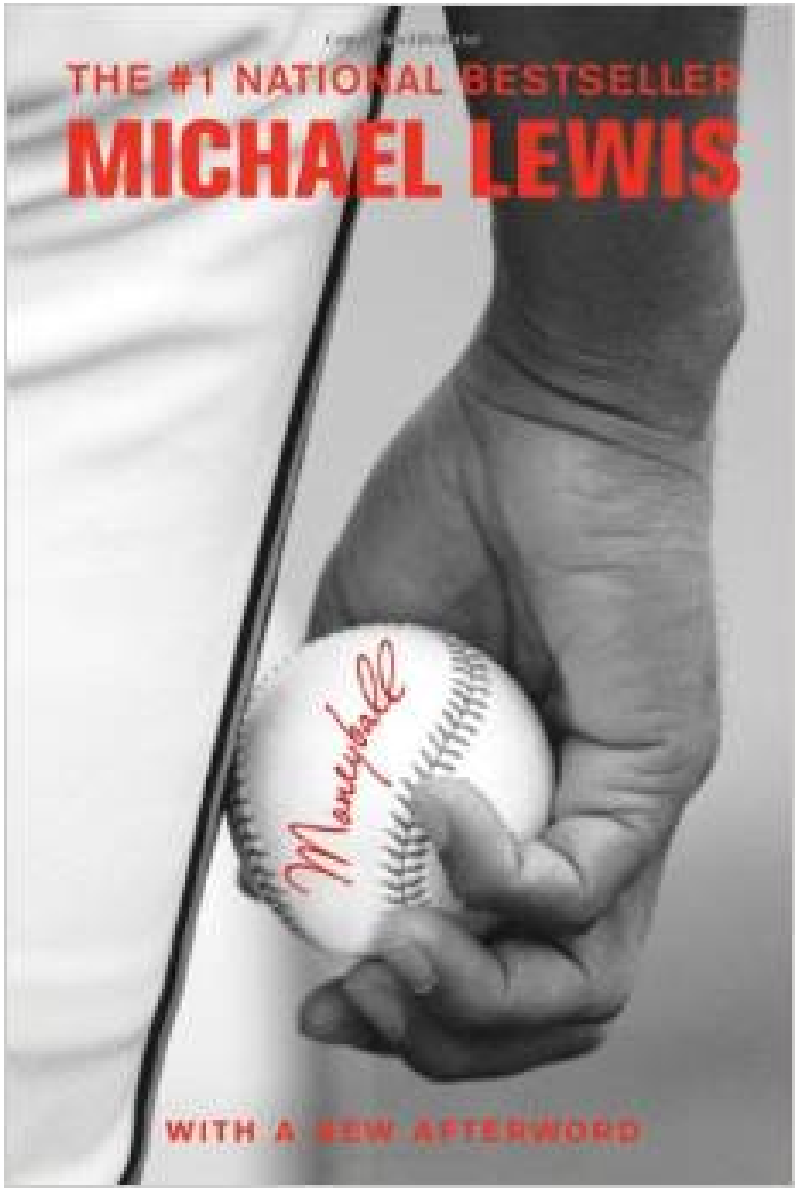





IBM Intelligent Operations software is designed with cities, for cities, to provide the tools to monitor, visualize and analyze vital city services such as water and wastewater systems, transportation, infrastructure planning, permit management and emergency response.

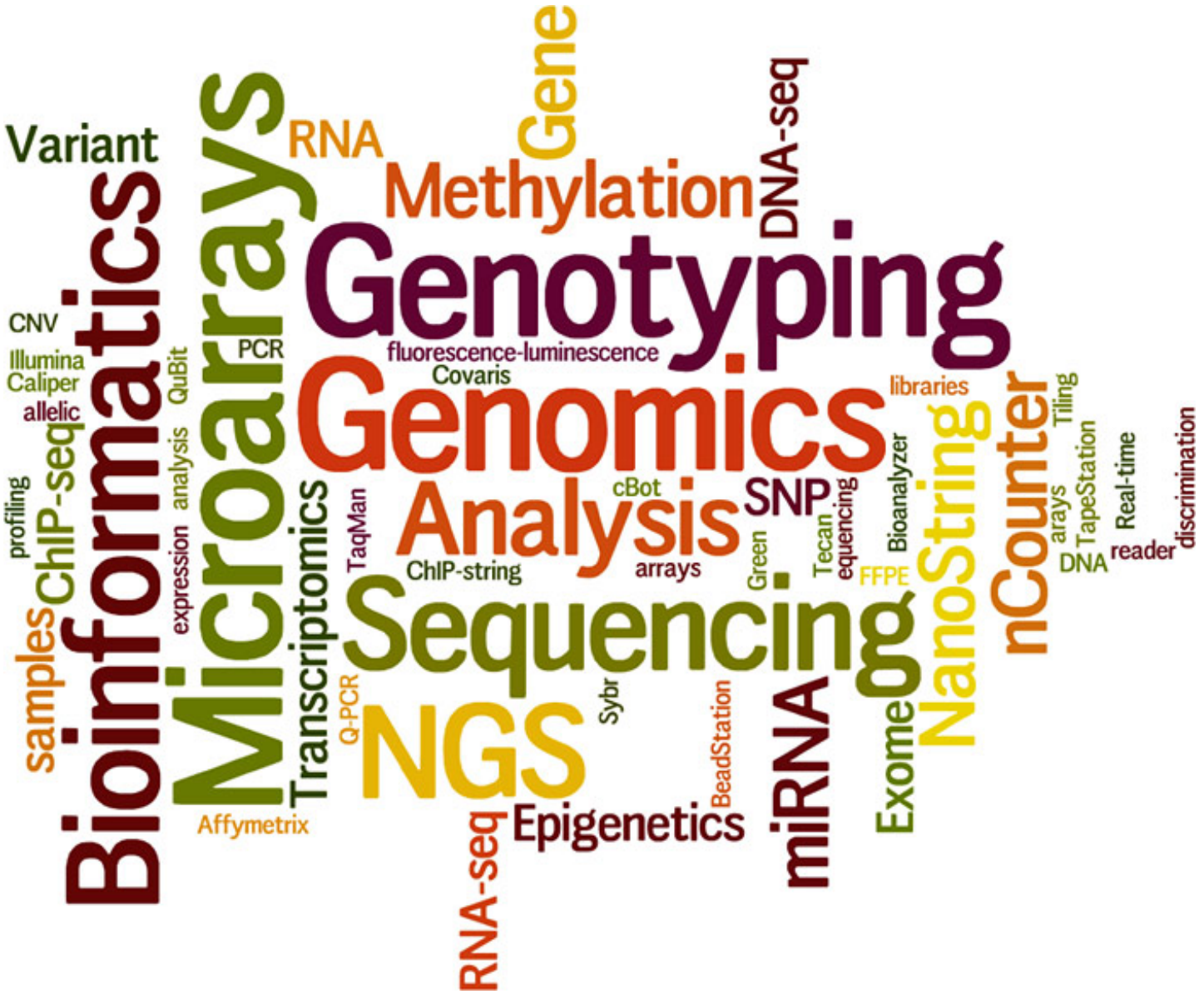
# From Data to Product

## Sports



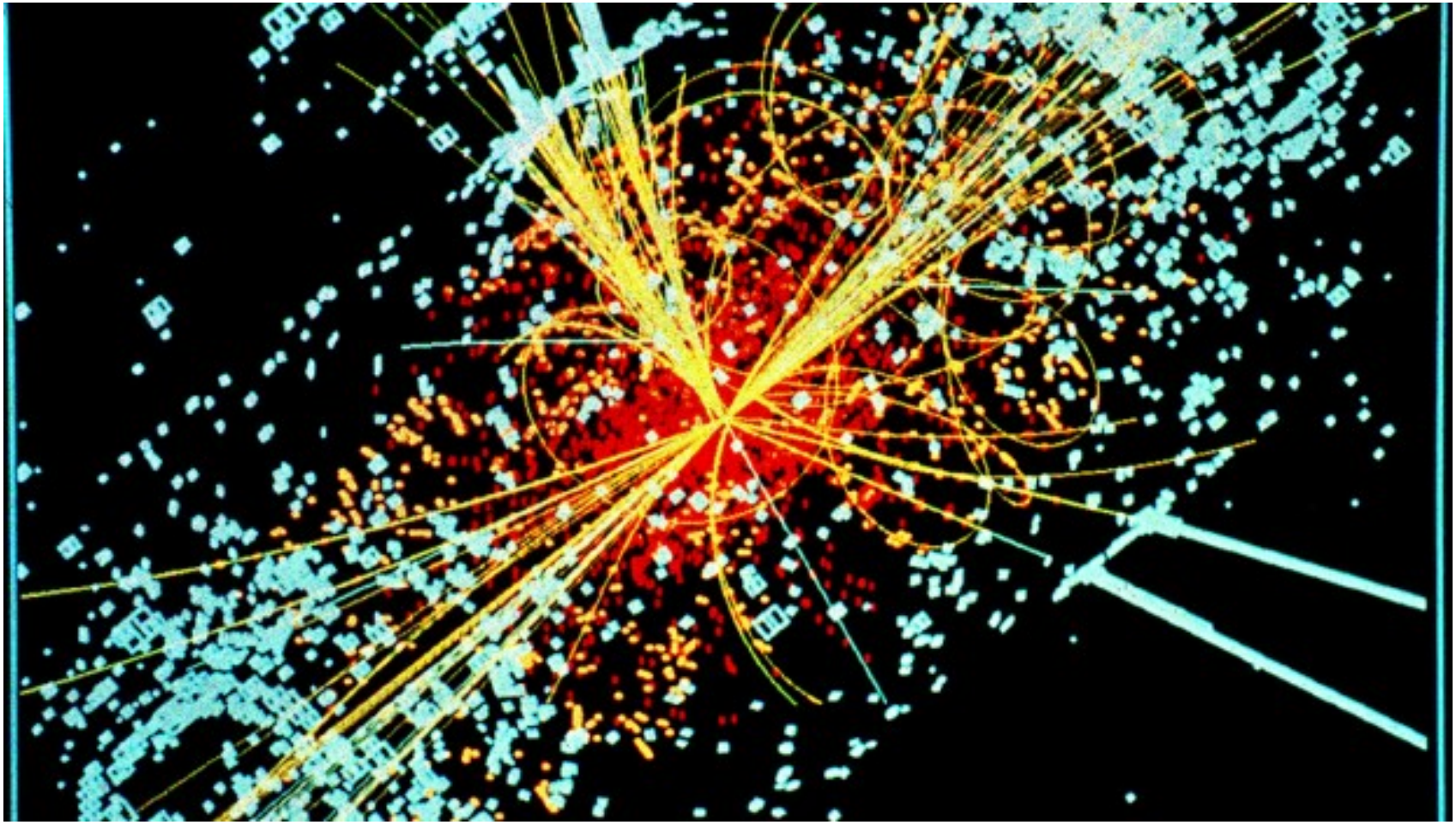
# From Data to Product

## Genomics



# From Data to Product

## Physics

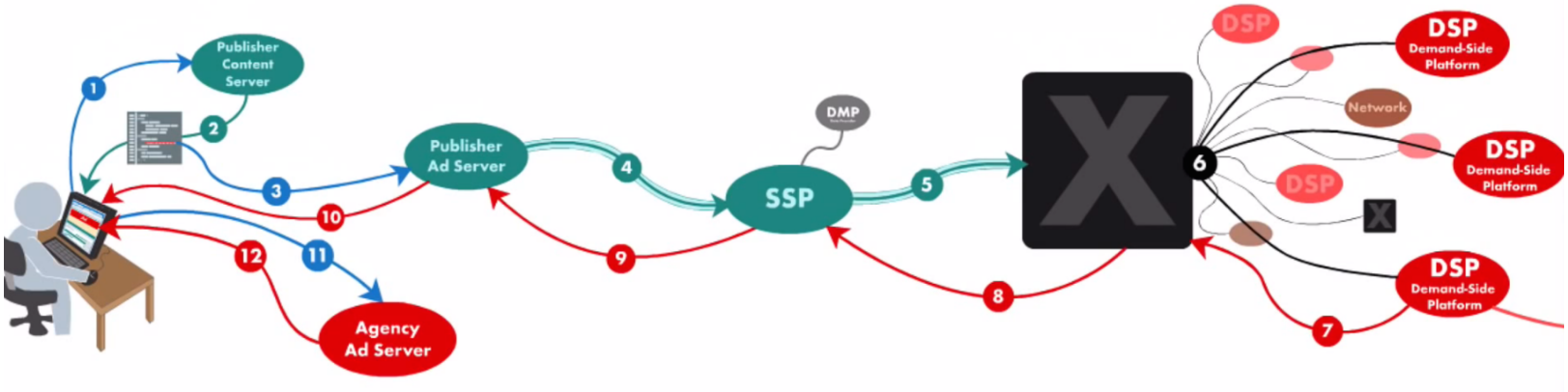


# An example: Real Time Bidding

## Outline

# An example: Real Time Bidding

An example: Real Time Bidding




- A **customer** visits a webpage with his browser: a complex process of content selection and delivery begins.

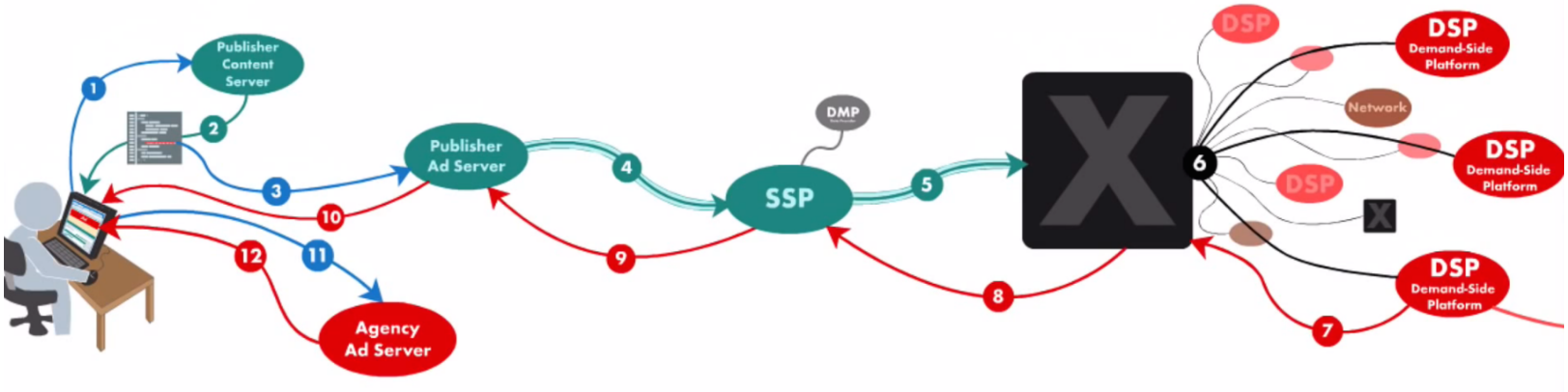
  An **advertiser** might want to display an ad to this customer on the webpage he is going to

  The webpage belongs to a **publisher**. The publisher delivers contents: news, music, information, sports, etc. This content draws an **audience**

  The publisher sells ad space to advertisers who want to reach that audience

# An example: Real Time Bidding

An example: Real Time Bidding



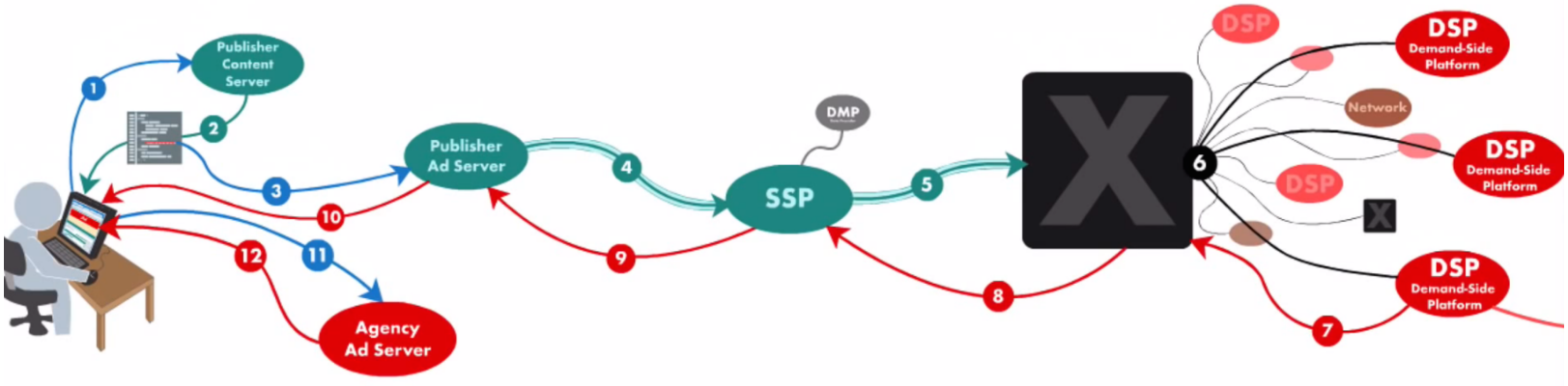

- **1. 2.** The customer visits a publisher's webpage: the browser opens a connection to the **publisher's content server**. It returns the content for the page (html code).

  The html code describing this content is retrieved by the browser, and it starts to render an interpret it.

  But... there is a line in this html code that says "follow this URL to retrieve ad content"

# An example: Real Time Bidding

An example: Real Time Bidding

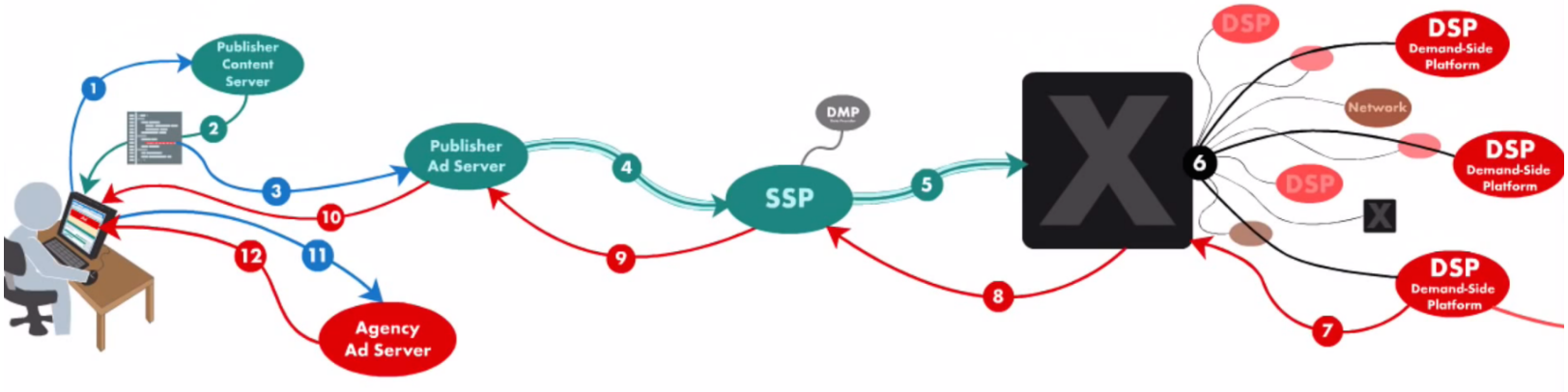



- **3.** The publisher has an **ad server**: it answers the request by considering possibilities: can I put an ad for my premium buyers ? Do I have data about this consumer viewing my content? (it could help me to decide to which buyer I could give this display opportunity). Only logical rules apply (no machine-learning here).

  The ad display opportunity is not premium, and this space or type of customer is not already reserved by a buyer. **Publisher's ad server** puts this opportunity of ad display in the open ad market.

# An example: Real Time Bidding

An example: Real Time Bidding



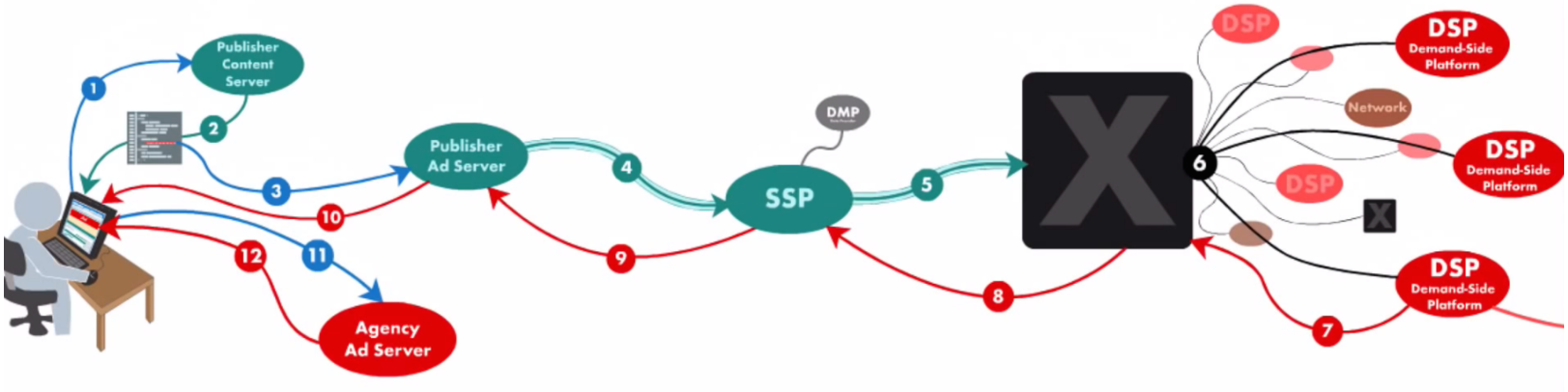

- **4.** The **publisher ad server** connects to an **SSP** (Supply-Side Platform). This platform monetizes its programmable display inventory.

  The SSP asks: have I already seen this consumer before ? Do I have additional data on him? The SSP requests extra information to a **DMP** (Data-Management Platform) about the user: profiling, audience segments, etc. Here machine learning is applied.

- **5.** Using this information, the SSP sends the ad request to an **ad-exchange**

# An example: Real Time Bidding

An example: Real Time Bidding



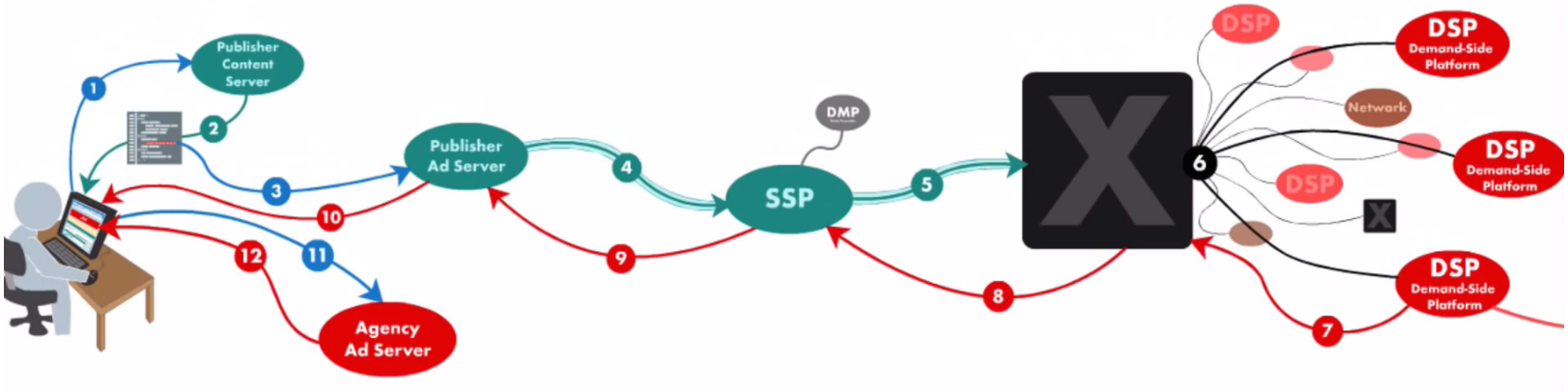

- **6.** Meanwhile, the ad-exchange is connected and exchanges with many potential buying systems: DSP (Demand-Side Platform), ad-networks, even other ad-exchange networks.

  Ad-network and DSP can have **pre-cached bid**: I'm paying 1$ for 1000 displays of 25years-old males in France, I buy 100 displays as soon as the price is below some threshold (like a broker).

  If no pre-cached bids, the ad-exchange says: no direct buyer for this display. Let's us an auction rule! The **RTB** (Real Time Bidding) begins.

# An example: Real Time Bidding

An example: Real Time Bidding

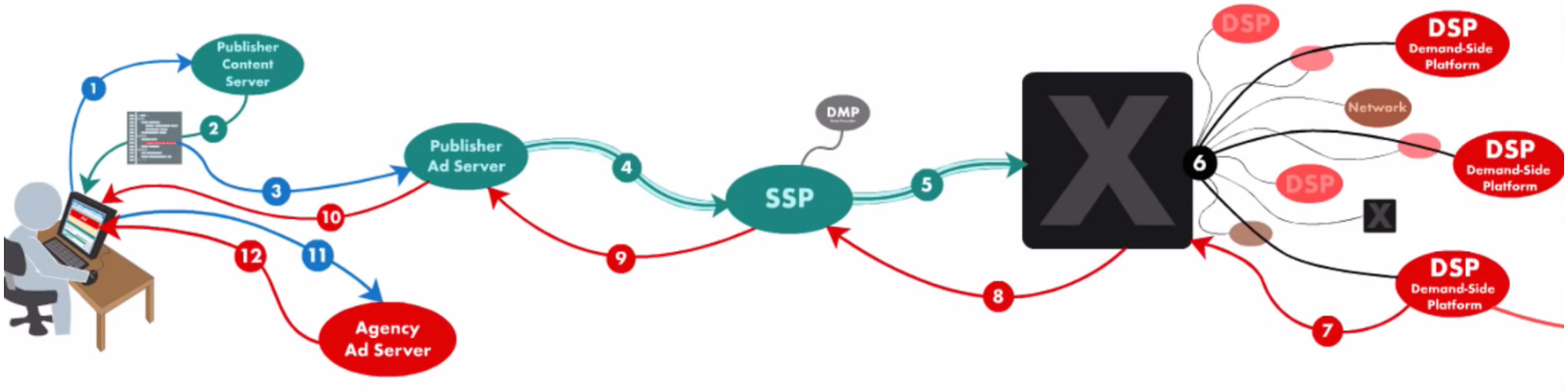



- **6. RTB:** buyers have **10ms** (!) to give a price to the ad-exchange. Buyers assess in real-time how willing they are to display an ad to this customer.

  Machine learning is used here, but only the **prediction** step, e.g. to assess the probability that the customer will click on some ads. The model must contain few parameters to answer quickly: the use of feature selection in the training step is crucial here.

# An example: Real Time Bidding

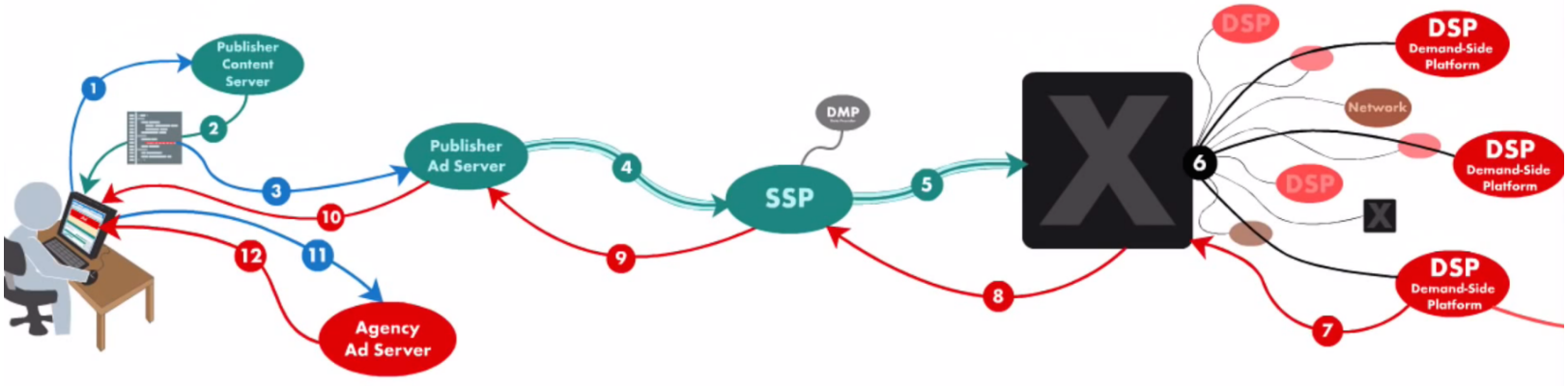
An example: Real Time Bidding




- **7.** The ad-exchange selects the highest bidder. The winning DSP gives instruction to the ad-exchange to retrieve the ad creative.
- **8.** The ad-exchange passes these instructions to the SSP
- **9.** The SSP send the request to the publisher ad server
- **10.** The publisher ad server responds to the still existing http connection of the browser,
- **11. 12.** and tells to the browser to go to the **agency's ad server** to download the ad.

# An example: Real Time Bidding

An example: Real Time Bidding



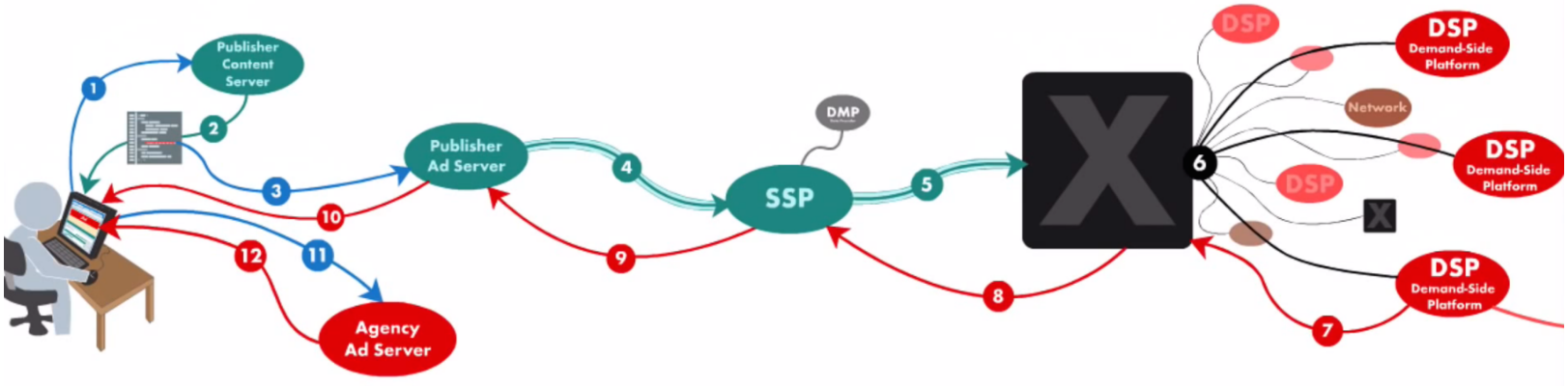

- Now the ad can be displayed in the browser.

Full process takes $< 100$ms !

- Where is data science:
  - DMP side, to cluster audience into marketing segments and to profile customers: clustering and classification
  - Buyer's side (DSP, ad-network) to compute the price proposed for RTB. Need to estimate the probability of a click on ads: regression and classification

# An example: Real Time Bidding

An example: Real Time Bidding




- Some numbers for a large web-advertisement company:
  - 10 million prediction of click probability per second
  - answers within 10ms
  - stores 20Terabytes of data daily

### Data everywhere

- Huge volume,
- Huge variety...

### Affordable computation units

- Cloud computing
- Graphical Processor Units (GPU)...

- Growing academic and industrial interest!

# Data Science ecosystem

## Big Data is (quite) Easy

### Example of *off the shelves* solution



```
def run(params: Params) {
  val conf = new SparkConf()
     .setAppName(s"BinaryClassification with $params")
  val sc = new SparkContext(conf)

  Logger.getRootLogger.setLevel(Level.WARN)

  val examples = MLUtils.loadLibSVMFile(sc, params.input).cache()

  val splits = examples.randomSplit(Array(0.8, 0.2))
  val training = splits(0).cache()
  val test = splits(1).cache()
  val numTraining = training.count()
  val numTest = test.count()
  println(s"Training: $numTraining, test: $numTest.")
  examples.unpersist(blocking = false)

  val updater = params.regType match {
    case L1 => new L1Updater()
    case L2 => new SquaredL2Updater()
  }

  val algorithm = new LogisticRegressionWithSGD()
     algorithm.optimizer
       .setNumIterations(params.numIterations)
       .setStepSize(params.stepSize)
       .setUpdater(updater)
       .setRegParam(params.regParam)
  val model = algorithm.run(training).clearThreshold()

  val prediction = model.predict(test.map(_.features))
  val predictionAndLabel = prediction.zip(test.map(_.label))

  val metrics = new BinaryClassificationMetrics(predictionAndLabel)
  val myMetrics = new MyBinaryClassificationMetrics(predictionAndLabel)

  println(s"Empirical CrossEntropy = ${myMetrics.crossEntropy()}.")
  println(s"Test areaUnderPR = ${metrics.areaUnderPR()}.")
  println(s"Test areaUnderROC = ${metrics.areaUnderROC()}.")

  sc.stop()
}
```

# Data Science ecosystem

## Big Data is (quite) Easy

### Example of *off the shelves* solution



```
export AWS_ACCESS_KEY_ID=<your-access-keyid>
export AWS_SECRET_ACCESS_KEY=<your-access-key-secret>
cellule/spark/ec2/sparl-ec2 -i cellule.pem -k cellule -s <number of machines> launch <cluster-name>
ssh -i cellule.pem root@<your-cluster-master-dns>
spark-ec2/copy-dir ephemeral-hdfs/conf
ephemeral-hdfs/bin/hadoop distcp s3n://celluledecalcul/dataset/raw/train.csv /data/train.csv
scp -i cellule.pem cellule/challenge/target/scala-2.10/target/scala-2.10/challenges_2.10-0.0.jar

cellule/spark/bin/spark-submit \
        --class fr.cc.challenge.Preprocess \
        challenges_2.10-0.0.jar \
        /data/train.csv \
        /data/train2.csv

cellule/spark/bin/spark-submit \
       --class fr.cc.sparktest.LogisticRegression \
       challenges_2.10-0.0.jar \
       /data/train2.csv
```

$\Rightarrow$ Logistic regression for arbitrary large dataset!

# Data Science ecosystem

## Big Data is (quite) Easy

### Example of *off the shelves* solution



```
def run(params: Params) {
  val conf = new SparkConf()
     .setAppName(s"BinaryClassification with $params")
  val sc = new SparkContext(conf)

  Logger.getRootLogger.setLevel(Level.WARN)

  val examples = MLUtils.loadLibSVMFile(sc, params.input).cache()

  val splits = examples.randomSplit(Array(0.8, 0.2))
  val training = splits(0).cache()
  val test = splits(1).cache()
  val numTraining = training.count()
  val numTest = test.count()
  println(s"Training: $numTraining, test: $numTest.")
  examples.unpersist(blocking = false)

  val updater = params.regType match {
    case L1 => new L1Updater()
    case L2 => new SquaredL2Updater()
  }

  val algorithm = new LogisticRegressionWithSGD()
     algorithm.optimizer
       .setNumIterations(params.numIterations)
       .setStepSize(params.stepSize)
       .setUpdater(updater)
       .setRegParam(params.regParam)
  val model = algorithm.run(training).clearThreshold()

  val prediction = model.predict(test.map(_.features))
  val predictionAndLabel = prediction.zip(test.map(_.label))

  val metrics = new BinaryClassificationMetrics(predictionAndLabel)
  val myMetrics = new MyBinaryClassificationMetrics(predictionAndLabel)

  println(s"Empirical CrossEntropy = ${myMetrics.crossEntropy()}.")
  println(s"Test areaUnderPR = ${metrics.areaUnderPR()}.")
  println(s"Test areaUnderROC = ${metrics.areaUnderROC()}.")

  sc.stop()
}
```

# Data Science ecosystem

## Big Data is (quite) Easy

### Example of *off the shelves* solution

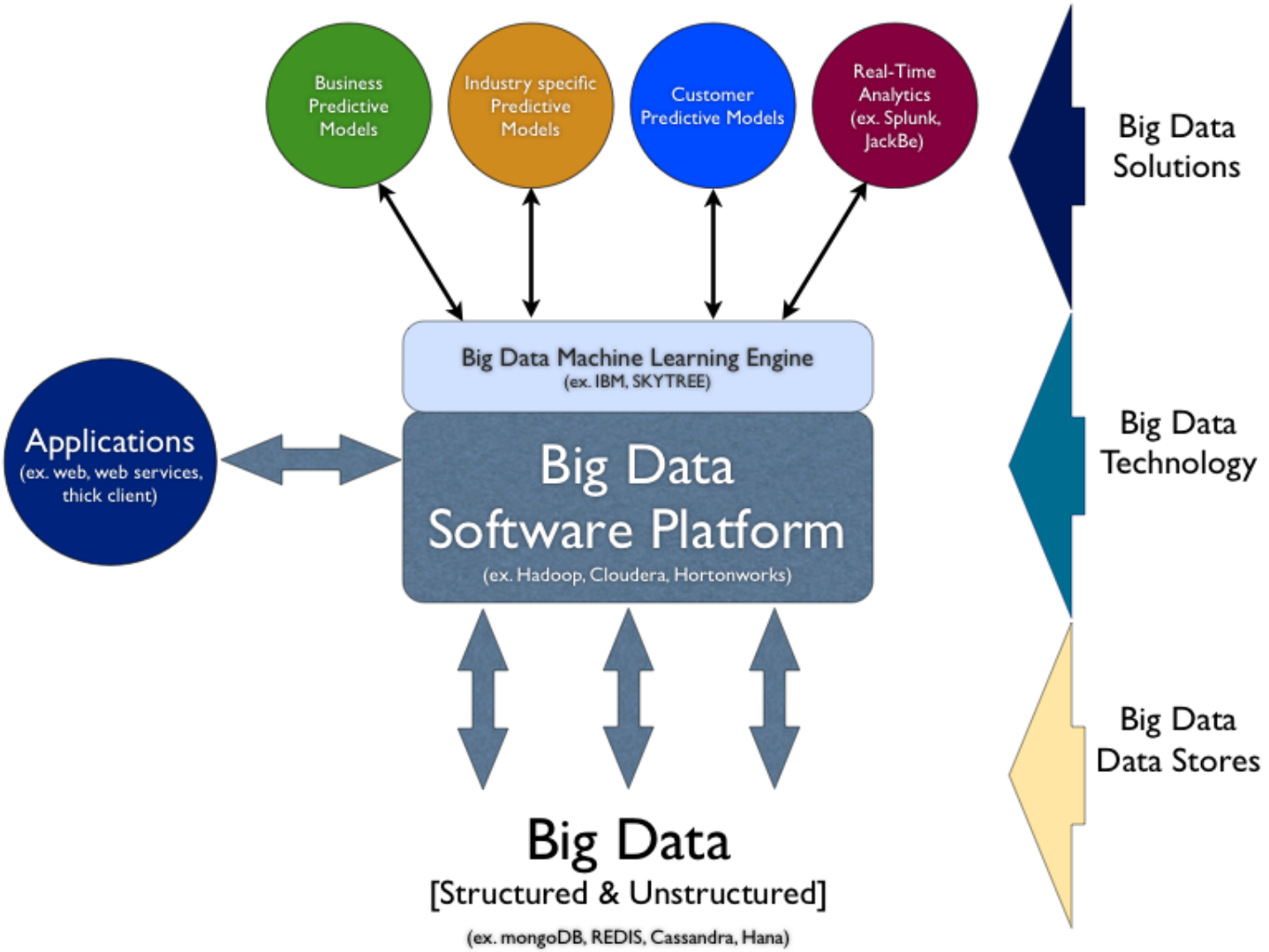


```
export AWS_ACCESS_KEY_ID=<your-access-keyid>
export AWS_SECRET_ACCESS_KEY=<your-access-key-secret>
cellule/spark/ec2/sparl-ec2 -i cellule.pem -k cellule -s <number of machines> launch <cluster-name>
ssh -i cellule.pem root@<your-cluster-master-dns>
spark-ec2/copy-dir ephemeral-hdfs/conf
ephemeral-hdfs/bin/hadoop distcp s3n://celluledecalcul/dataset/raw/train.csv /data/train.csv
scp -i cellule.pem cellule/challenge/target/scala-2.10/target/scala-2.10/challenges_2.10-0.0.jar

cellule/spark/bin/spark-submit \
        --class fr.cc.challenge.Preprocess \
        challenges_2.10-0.0.jar \
        /data/train.csv \
        /data/train2.csv

cellule/spark/bin/spark-submit \
       --class fr.cc.sparktest.LogisticRegression \
       challenges_2.10-0.0.jar \
       /data/train2.csv
```

$\Rightarrow$ Logistic regression for arbitrary large dataset!

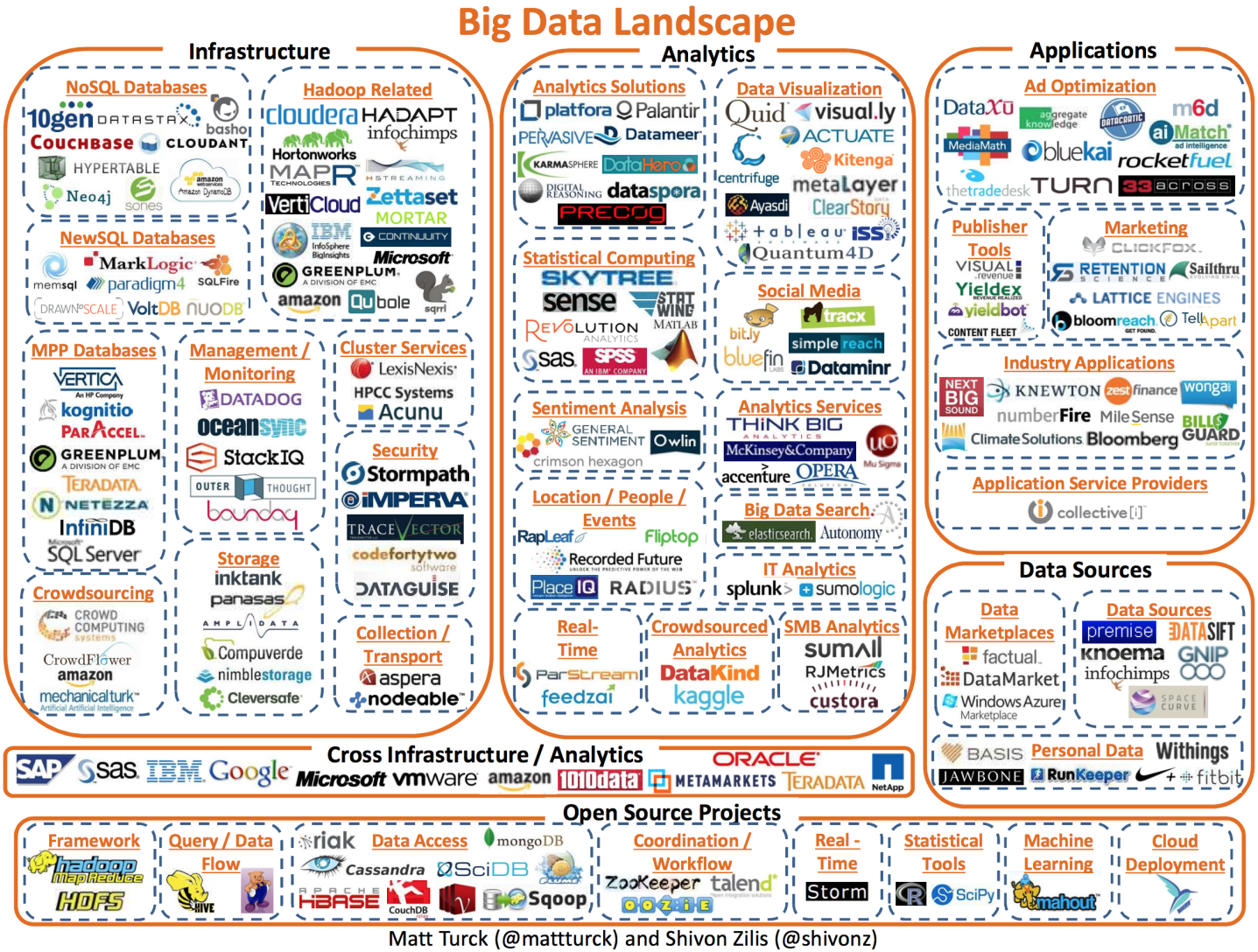
# Data Science ecosystem

## A Complex Ecosystem!



# Data Science ecosystem

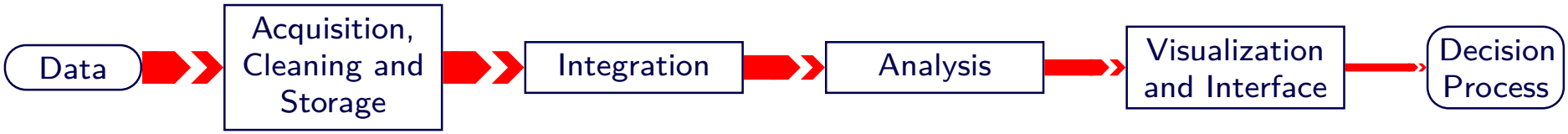
## A Complex Ecosystem!




Matt Turck (@mattturck) and Shivon Zilis (@shivonz)

# Data cycle
## Outline

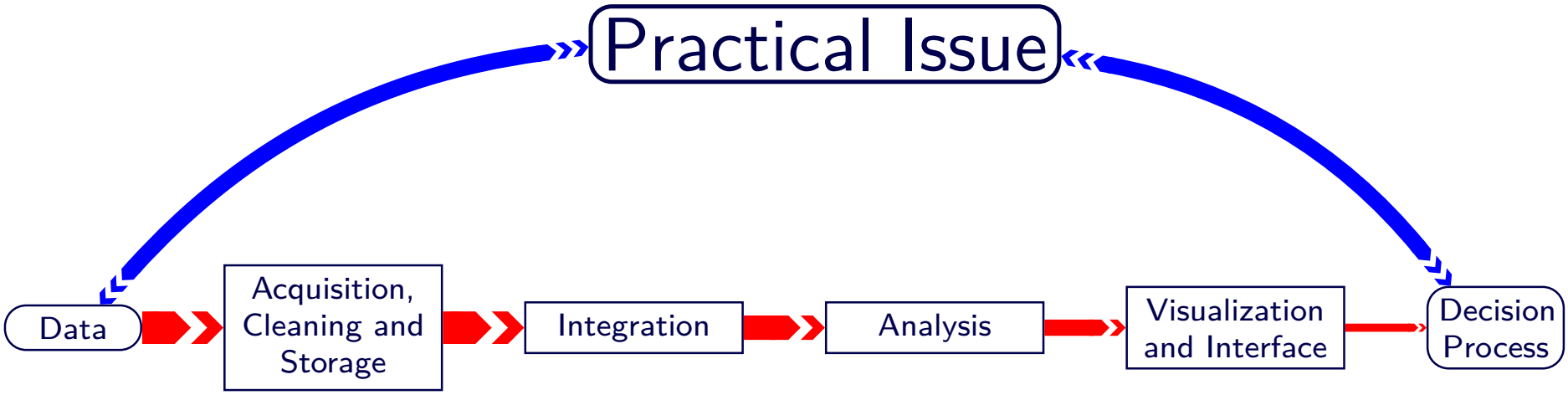
# Data cycle
## Data Cycle




- Data/Information flow vision

# Data cycle

## Data Cycle




- Data/Information flow vision
- Goal oriented

- Raw material:
  - Structured and unstructured data (**V**ariety)
  - Data quality issue (**V**eracity)
  - Quantity (**V**olume and **V**elocity)
- Various sources:
  - Open data,
  - Proprietary data

# Data cycle

## Acquisition, Cleaning, Storage and Integration

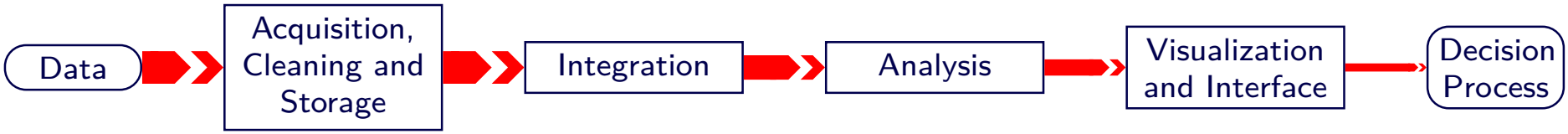
- Get the data from the sources.
- Storage issue and availability for processing.
- Cleaning and formating
- Integration: Data preparation for analysis
- Time consuming!

- Extract information from the data
- Statistics/Machine learning
- Big Data: hardware is the limit (time/volume)

- Reporting part: **V**isualization, text...
- Also used for data exploration
- Very important aspect!

- Better decisions: **V**alue
- Need to answer a problem/question!
- Need to formalize the problem: no answer without a question!
- Feedback everywhere...

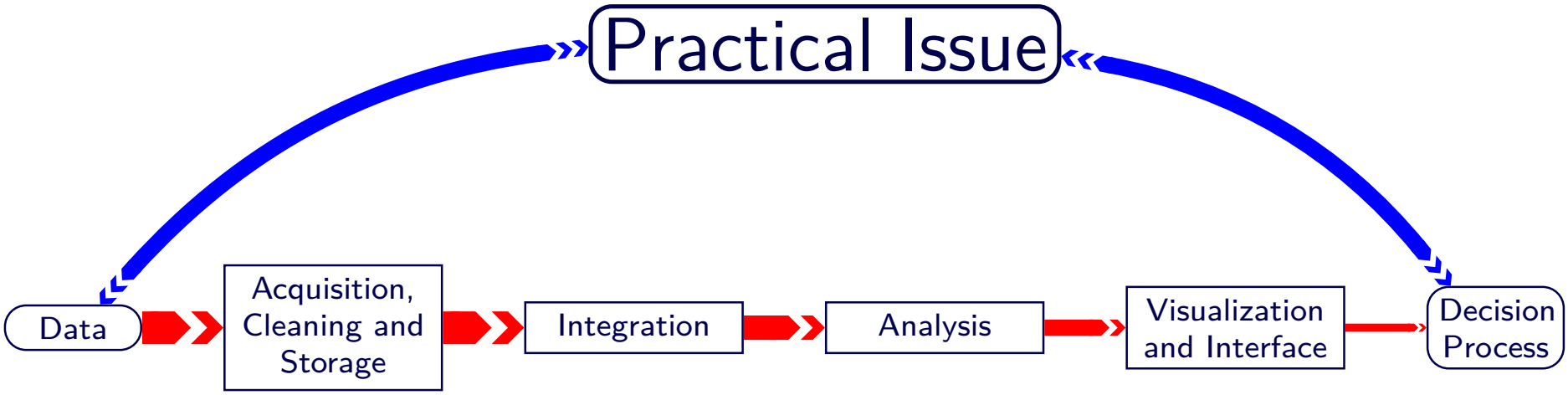
# Data cycle
## Real Data Cycle




- Data/Information flow vision

# Data cycle

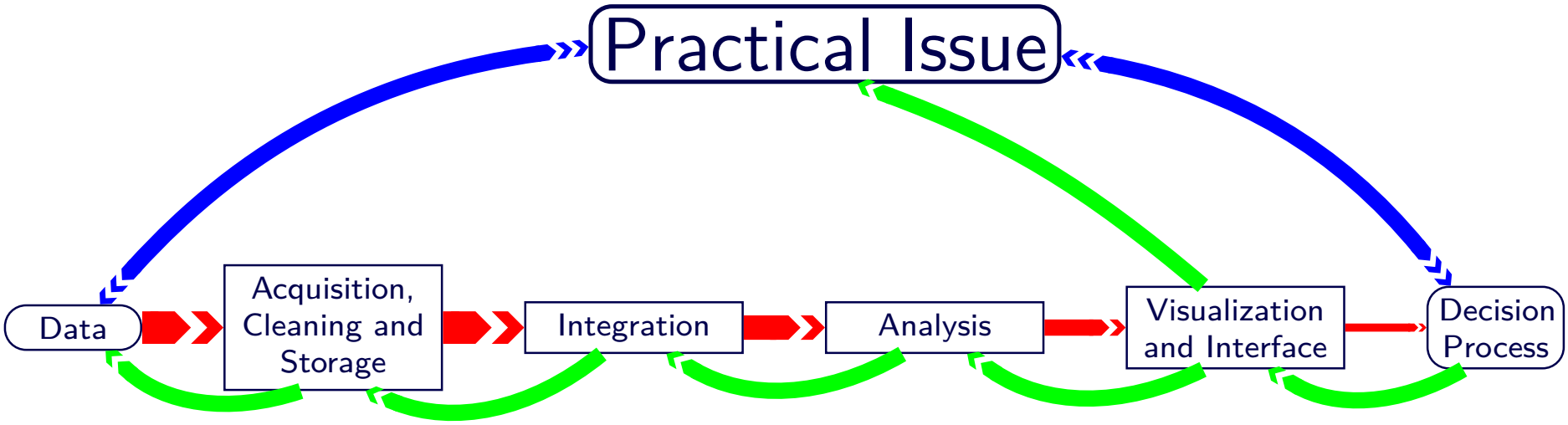
## Real Data Cycle




- Data/Information flow vision
- Goal oriented

# Data cycle

## Real Data Cycle




- Data/Information flow vision
- Goal oriented
- Iterative and interactive process

# Data cycle
## Doing Data Science

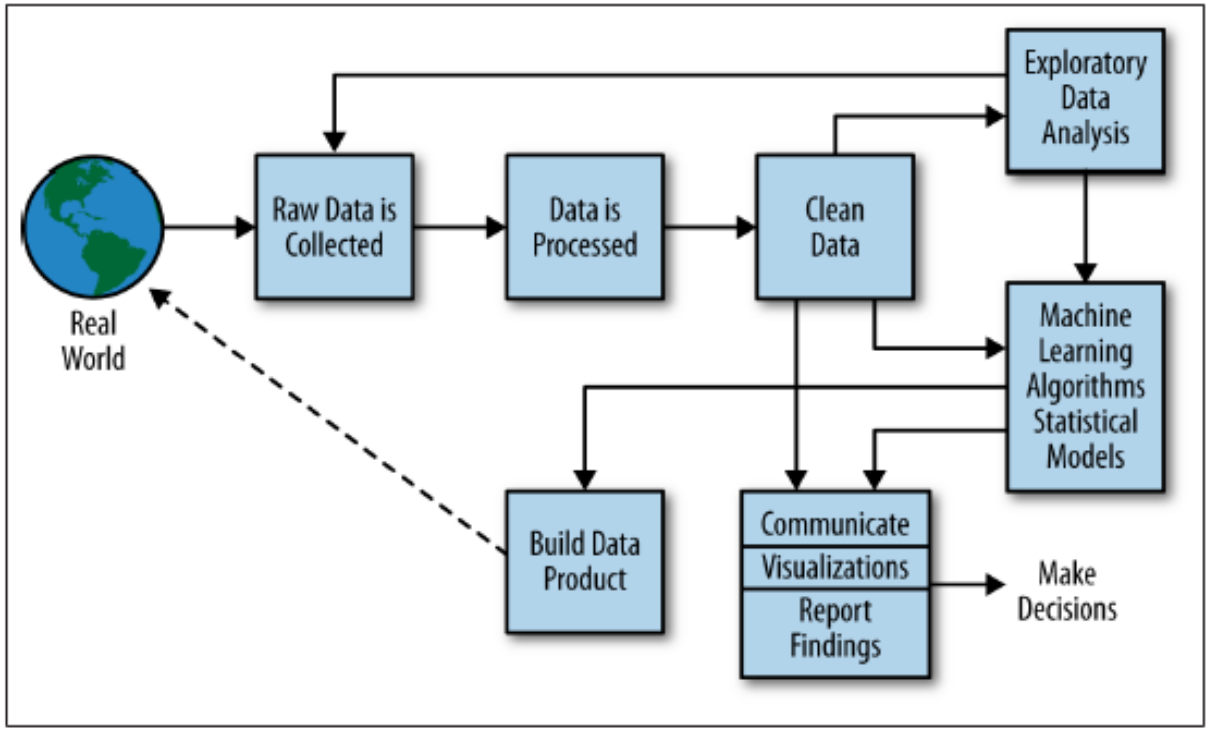



*Figure 2-2. The data science process*

- Doing Data Science: Straight talk from the frontline.
  - Rachel Schutt, Cathy O'Neil
  - O'Reilly

# Data Science project

## Outline

# Data Science project

## A 7 step program


1. Identify the problem
   - Type of problems and metric used to measure success
   - Identify key people within your organization and outside
   - Get specifications, requirements, priorities, budgets
   - How accurate the solution needs to be?
   - Do we need all the data?
   - Outsourcing?


2. Identify available data sources
   - Extract and check sample data / Perform Exploratory Data Analysis
   - Assess quality of data, and value available in data
   - Identify data glitches, find work-around
   - Data quality improvement?
   - Verify with field expert that you understand the data
   - Infrastructure?

# Data Science project

## A 7 step program


2. Identify available data sources
    - Extract and check sample data / Perform Exploratory Data Analysis
    - Assess quality of data, and value available in data
    - Identify data glitches, find work-around
    - Data quality improvement?
    - Verify with field expert that you understand the data
    - Infrastructure?
3. Identify if additional data sources are needed
    - What? How much? How to?
    - Real time?
    - Do we need experimental design?


4. Data preparation and analyses
   - Data preparation and cleaning
   - Explore methodologies
   - Select variables and models
   - Detect / remove outliers
   - Validate chosen methodology
   - Measure accuracy, provide confidence intervals
   - Provide visualization and ask for feedback

# Data Science project

## A 7 step program


4. Data preparation and analyses
    - Data preparation and cleaning
    - Explore methodologies
    - Select variables and models
    - Detect / remove outliers
    - Validate chosen methodology
    - Measure accuracy, provide confidence intervals
    - Provide visualization and ask for feedback
5. Implementation, development
    - FSSRR: Fast, simple, scalable, robust, re-usable
    - Debugging
    - Need to create an API to communicate with other apps?


6. Communicate results
    - Integration and visualization
    - Discuss potential improvements (with cost estimates)
    - Provide training
    - Code and methodology documentation

# Data Science project

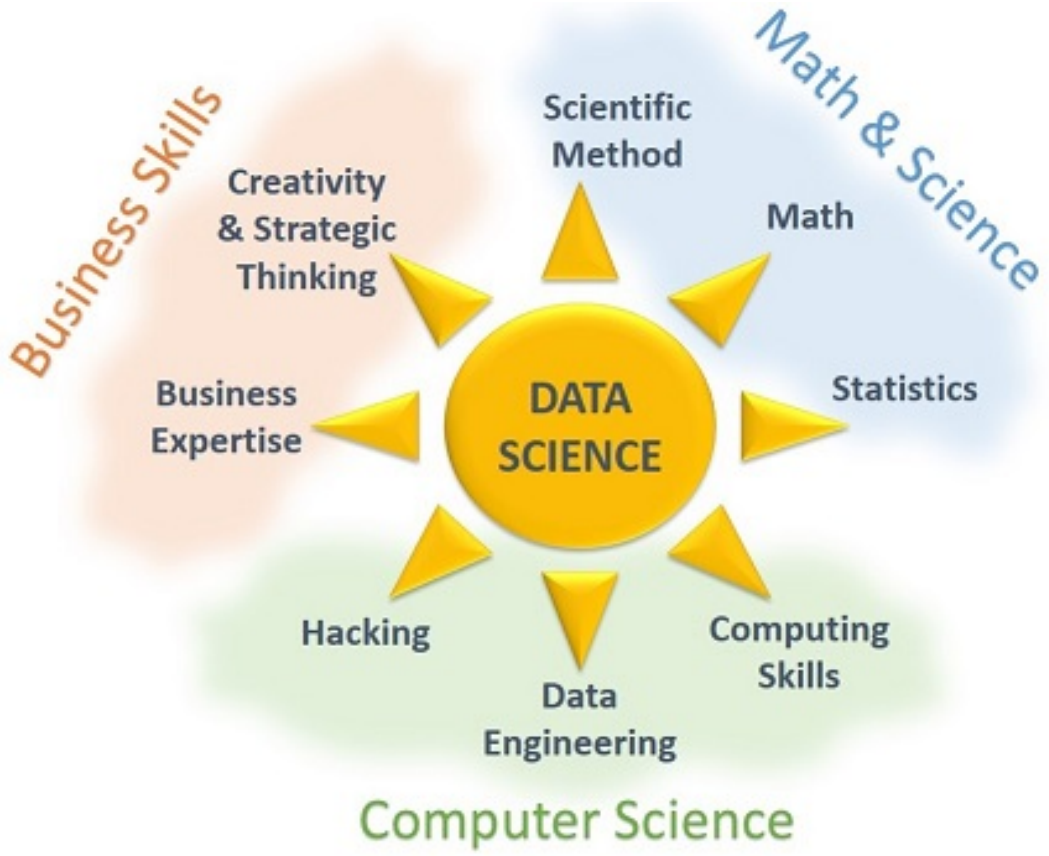
## A 7 step program


6. Communicate results
    - Integration and visualization
    - Discuss potential improvements (with cost estimates)
    - Provide training
    - Code and methodology documentation
7. Maintenance
    - Test the model or implementation; stress tests
    - Regular updates
    - Outsourcing?

- Business:
  - Business analysis, market knowledge, product usage, ...
- Data Management:
  - Data collection, storage, cleaning, filtering, integration,...
- Statistic and Machine Learning:
  - Data modeling, inference, prediction, pattern recognition,...
- Programming:
  - Software development, Large-scale or parallel data processing,...
- Interface and Data Visualization:
  - HCI design, visualization, story-telling,...

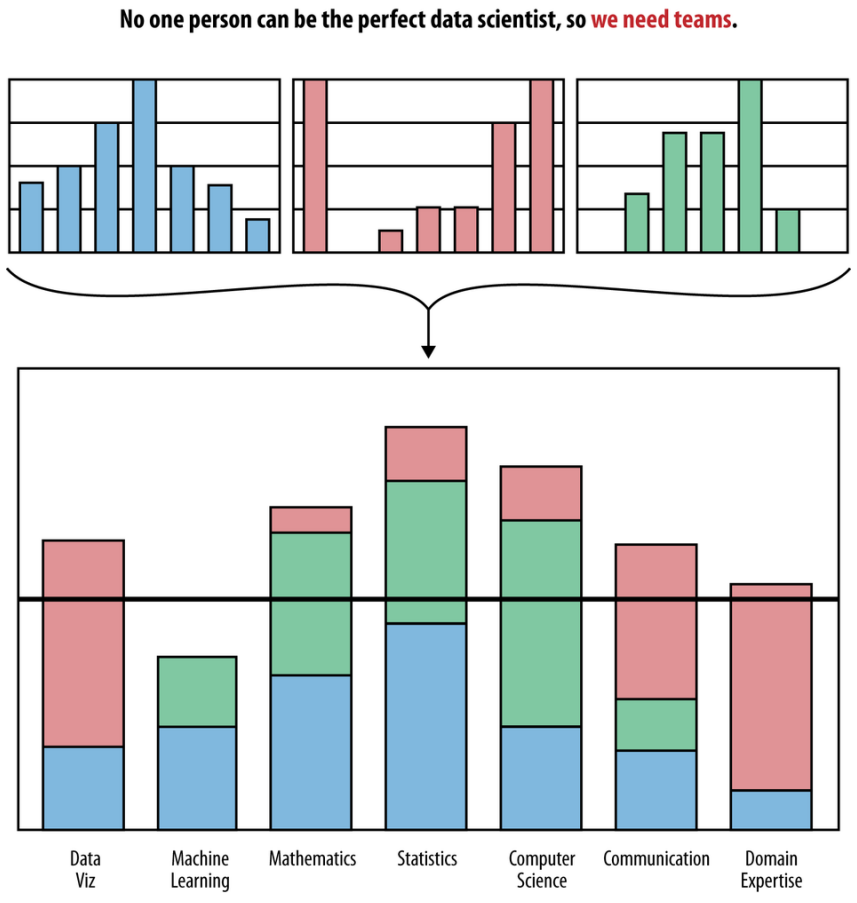
# Data scientists

## Profiles




- No one masters all the skills!

# Data scientists
## Data science team




- Gather people having different skills

There are the ones...

- Strong in **statistics**: develop new statistical theories for big data: statistical modeling, experimental design, sampling, clustering, data reduction, confidence intervals, testing, modeling, predictive modeling, etc.
- Strong in **mathematics**: operations research, analytic business (inventory management and forecasting, pricing optimization, supply chain, quality control, yield optimization)
- Strong in **data engineering**, Hadoop, database/memory/file systems optimization and architecture, API's, Analytics as a Service, optimization of data flows

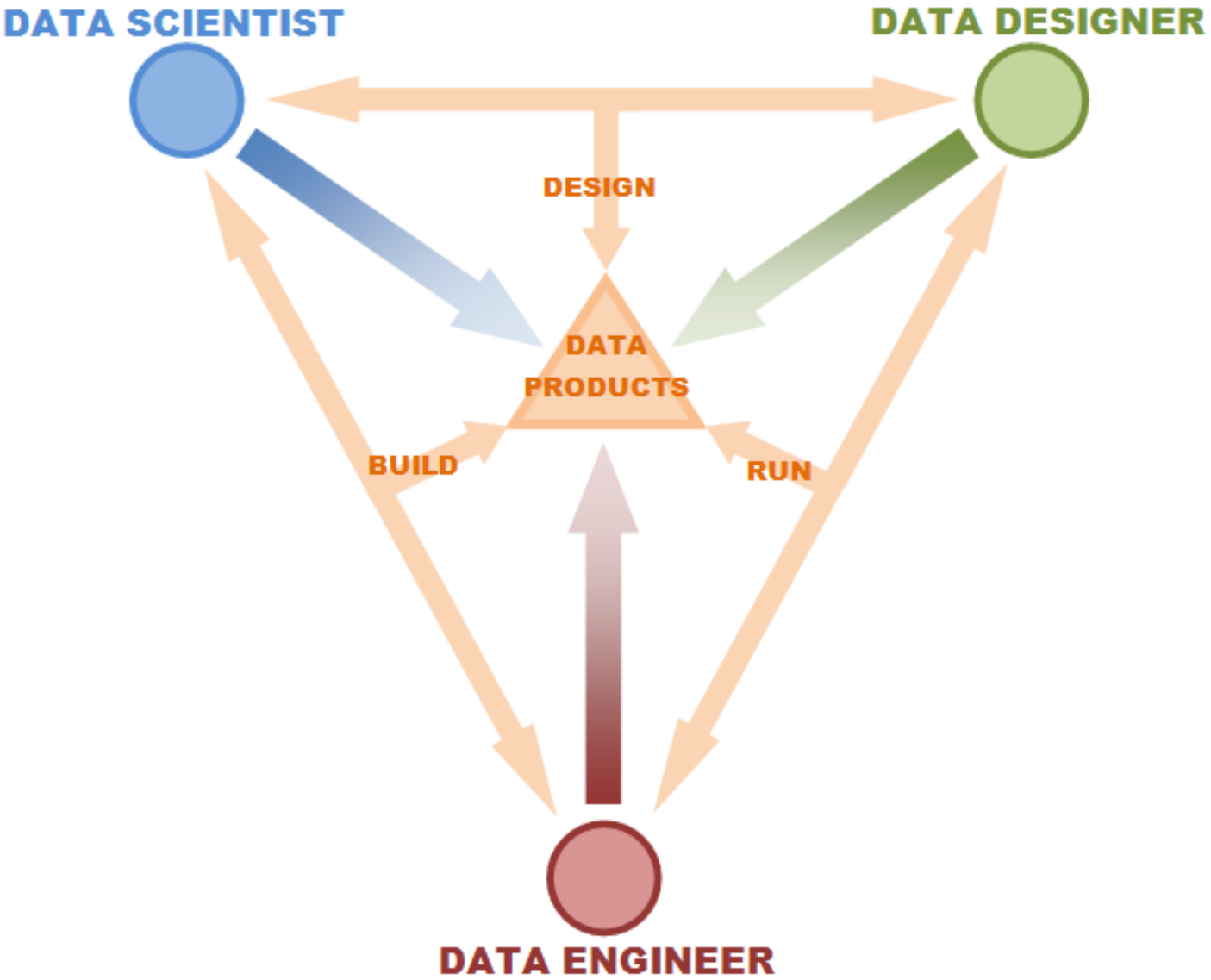
# Data scientists

## Main types of data scientists

- Strong in **computer science** (algorithms, computational complexity, optimization)
- Strong in **business**, ROI optimization, decision sciences (dashboards design, metric mix selection and metric definitions, ROI optimization, high-level database design)
- Strong in **production code** development, software engineering

# Data scientists

## More than data scientists?



# Big Data, Data Science, Statistics

## Outline

# Big Data, Data Science, Statistics

## Wikipedia

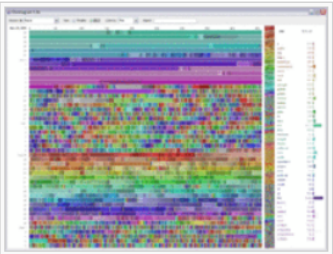
## Big data

From Wikipedia, the free encyclopedia

*This article is about large collections of data. For the band, see Big Data (band).*

**Big data**[1][2] is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications. The challenges include capture, curation, storage,[3] search, sharing, transfer, analysis[4] and visualization. The trend to larger data sets is due to the additional information derivable from analysis of a single large set of related data, as compared to separate smaller sets with the same total amount of data, allowing correlations to be found to "spot business trends, determine quality of research, prevent diseases, link legal citations, combat crime, and determine real-time roadway traffic conditions."[5][6][7]

As of 2012, limits on the size of data sets that are feasible to process in a reasonable amount of time were on the order of exabytes of data.[8] Scientists regularly encounter limitations due to large data sets in many areas, including meteorology, genomics,[9] connectomics, complex physics simulations,[10] and biological and environmental research.[11] The limitations also affect Internet search, finance and business informatics. Data sets grow in size in part because they are increasingly being gathered by ubiquitous information-sensing mobile devices, aerial sensory technologies (remote sensing), software logs, cameras, microphones, radio-frequency identification readers, and wireless sensor networks.[12][13] The world's technological per-capita capacity to store information has roughly doubled every 40 months since the 1980s;[14] as of 2012, every day 2.5 exabytes ($2.5 \times 10^{18}$) of data were created.[15] The challenge for large enterprises is determining who should own big data initiatives that straddle the entire organization.[16]



A visualization created by IBM of Wikipedia edits. At multiple terabytes in size, the text and images of Wikipedia are a classic example of big data.

Big data is difficult to work with using most relational database management systems and desktop statistics and visualization packages, requiring instead "massively parallel software running on tens, hundreds, or even thousands of servers".[17] What is considered "big data" varies depending on the capabilities of the organization managing the set, and on the capabilities of the applications that are traditionally used to process and analyze the data set in its domain. "For some organizations, facing hundreds of gigabytes of data for the first time may trigger a need to reconsider data management options. For others, it may take tens or hundreds of terabytes before data size becomes a significant consideration."[18]

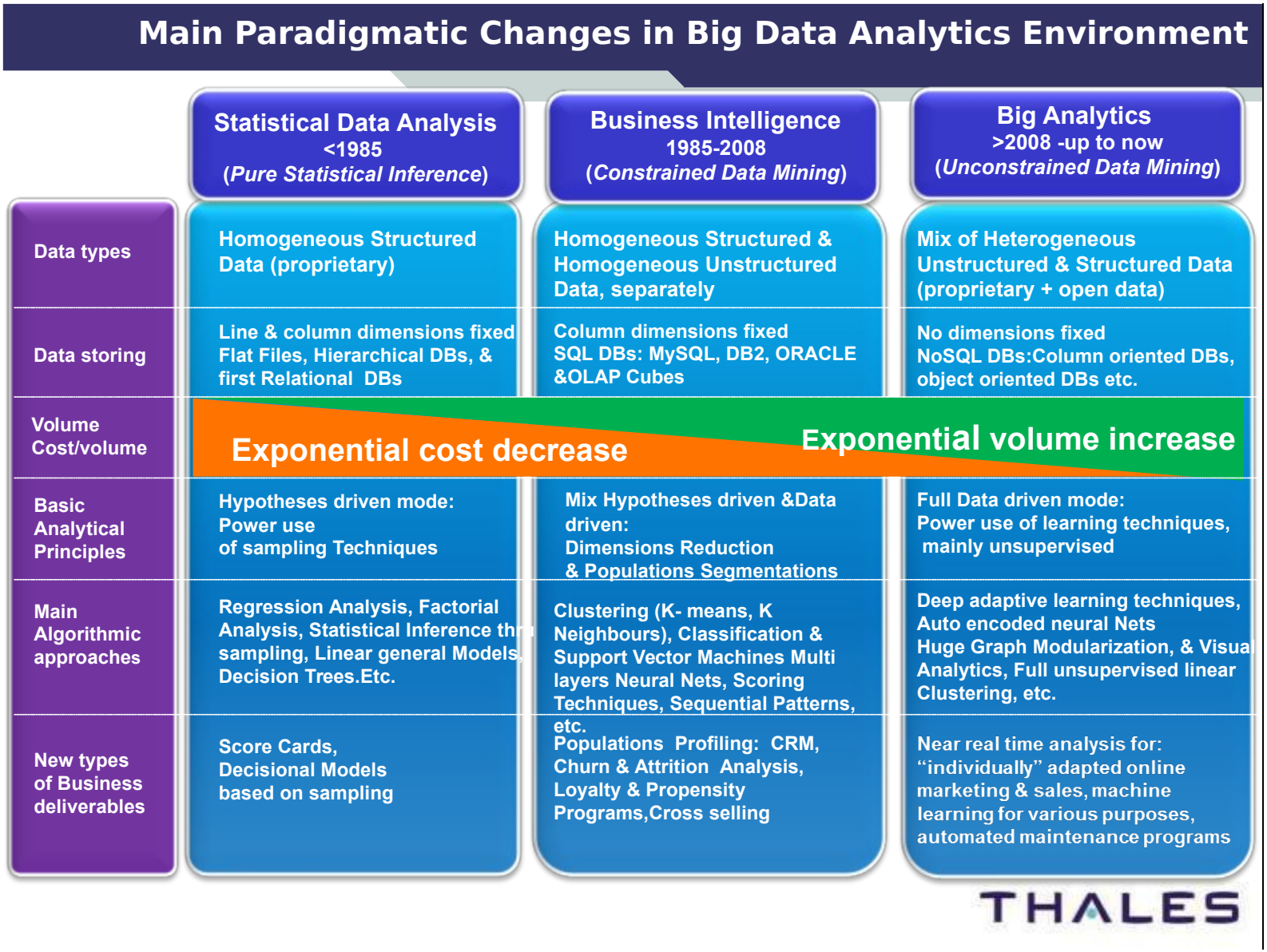
# Big Data, Data Science, Statistics

## Wikipedia

- **Big data** is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process using traditional data processing applications.
- **Data science** is the study of the generalizable extraction of knowledge from data, yet the key word is science.
- **Statistics** is the study of the collection, analysis, interpretation, presentation and organization of data.

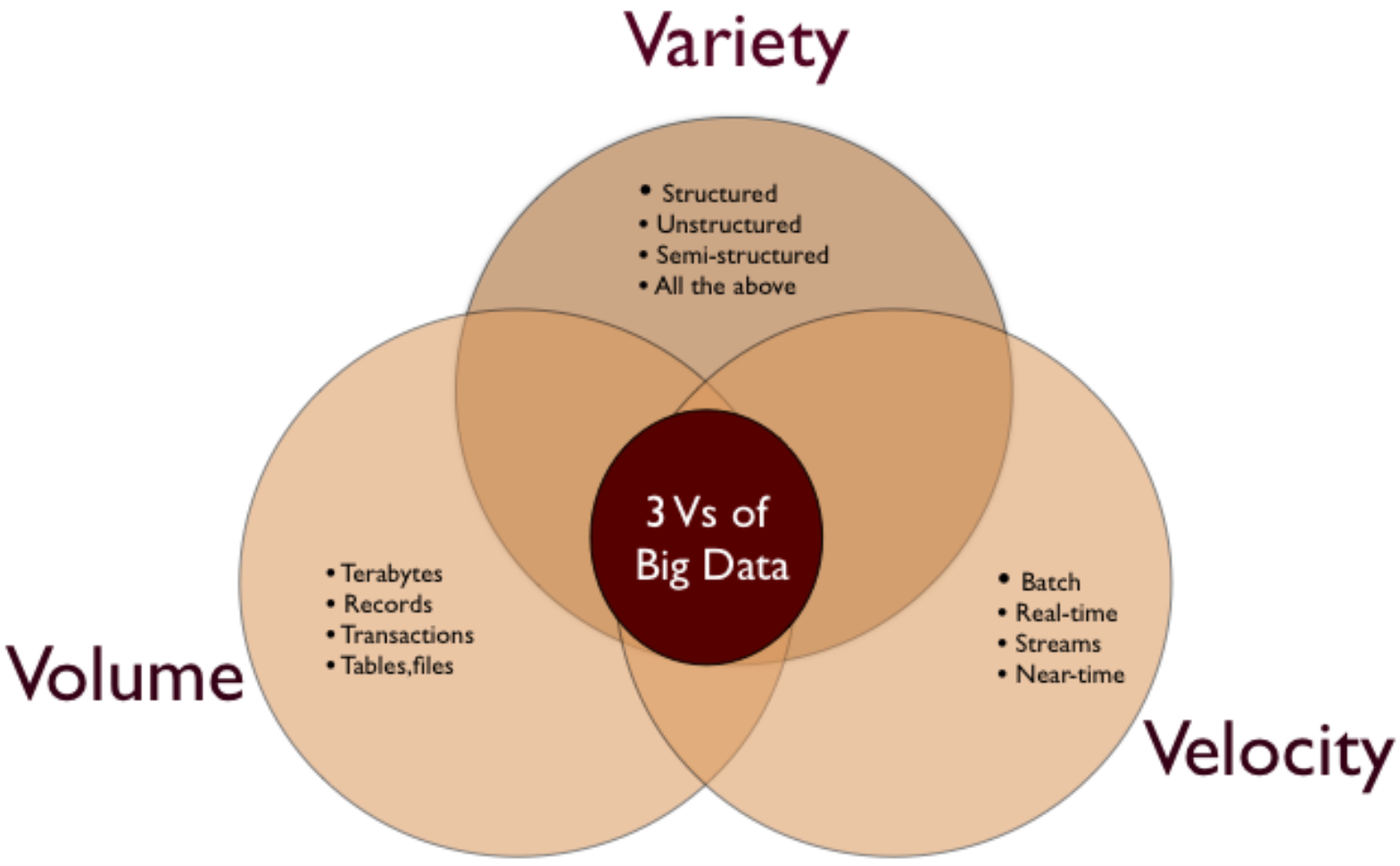
# Big Data, Data Science, Statistics

## Data science evolution

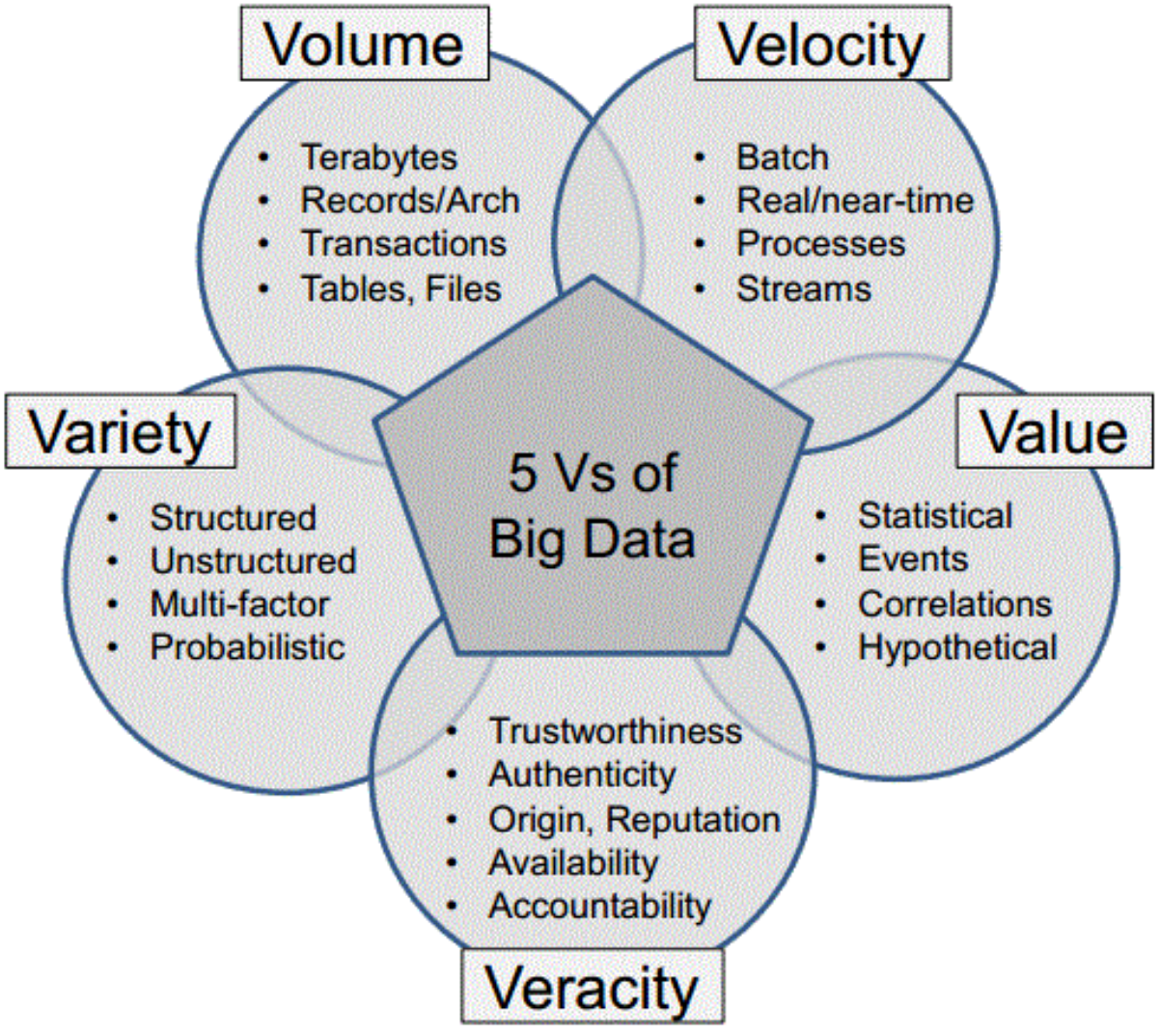


# Big Data, Data Science, Statistics

## 3Vs of Big Data



## 5Vs of Big Data

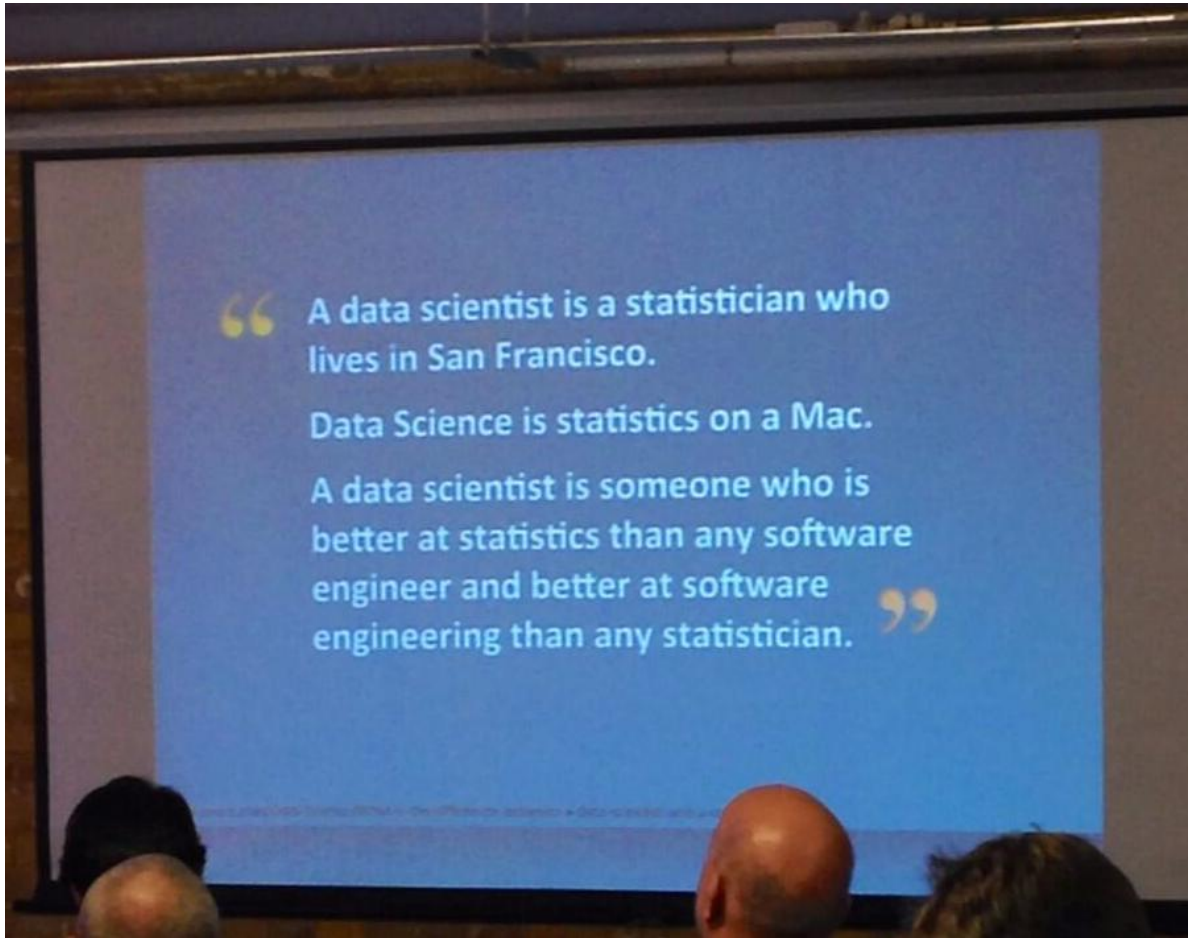


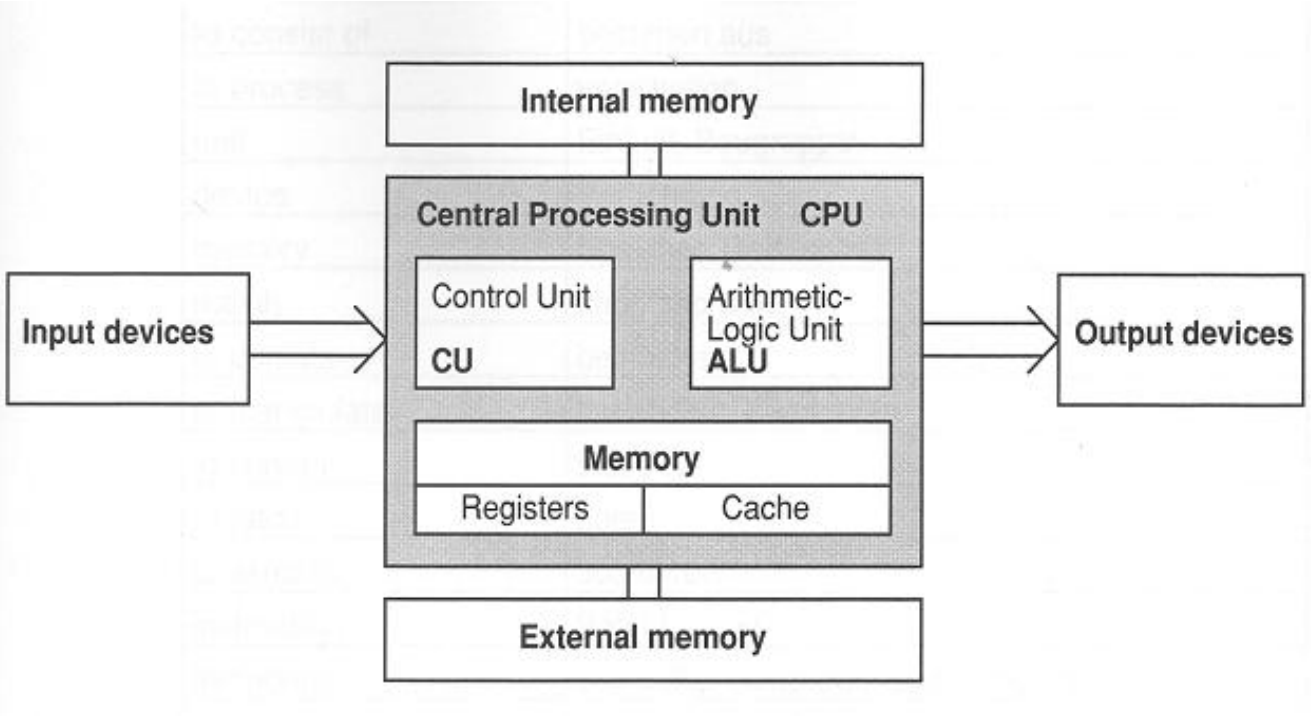
A vocabulary problem:

data scientist or statistician?

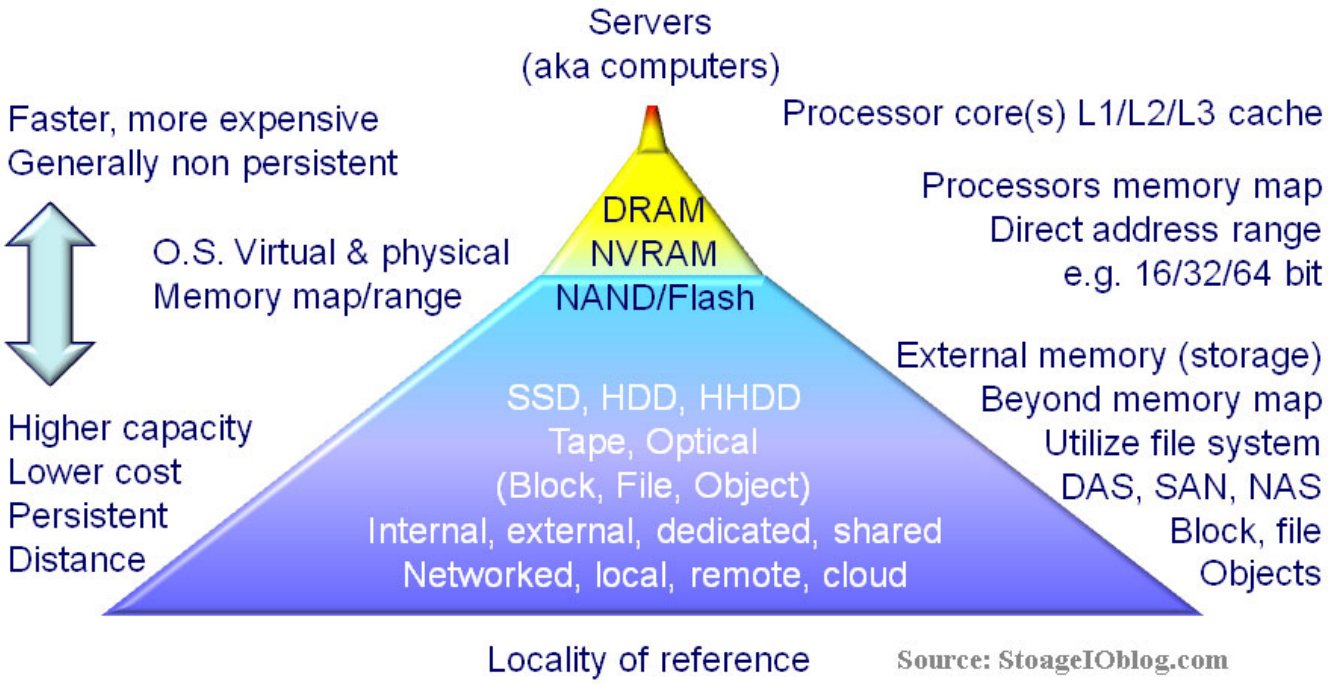
statistics or data science?

A possible answer:



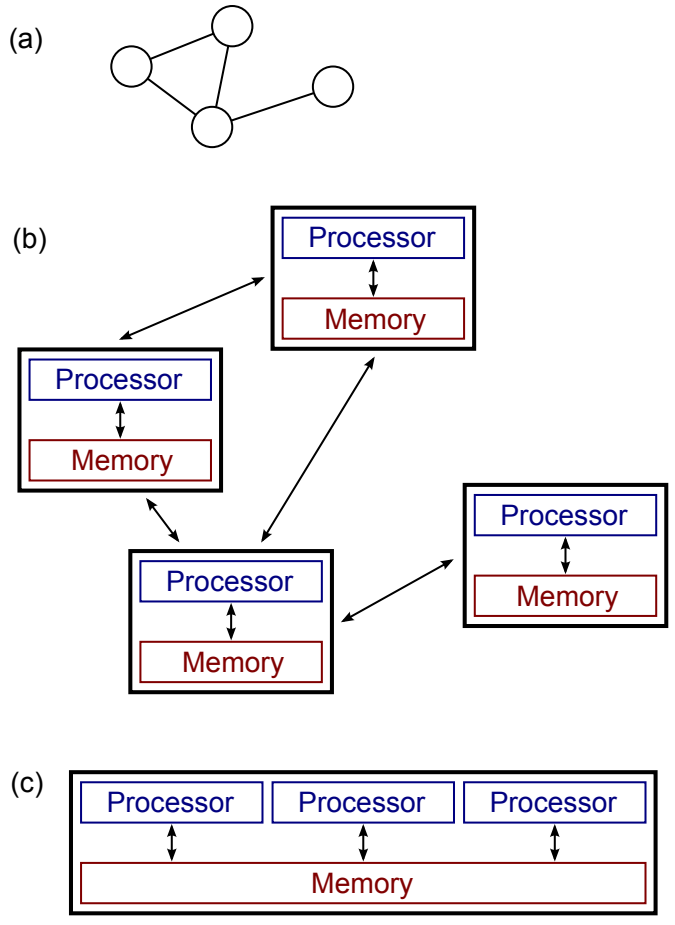



- Everything should go through the CPU...

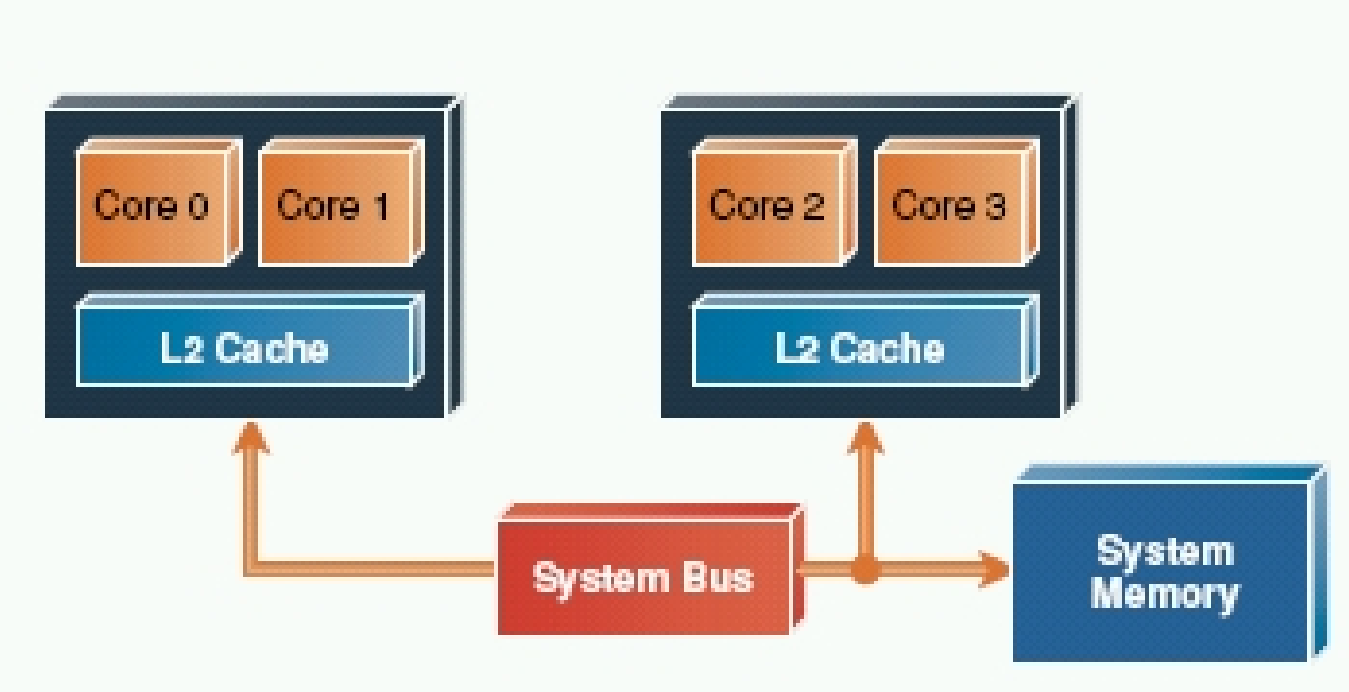



| | |
|---|---|
| CPU register | 64 b $\times$ 16 |
| Level 1 cache access | 8-128 kb |
| Level 2 cache access | 32-1024 kb |
| Level 3 cache access | 1-8 MB |
| Main memory access | 2-16 GB |
| Solid-state disk I/O | 250 GB-1 TB (4TB) |
| Rotational disk I/O | 500 GB-4 TB |

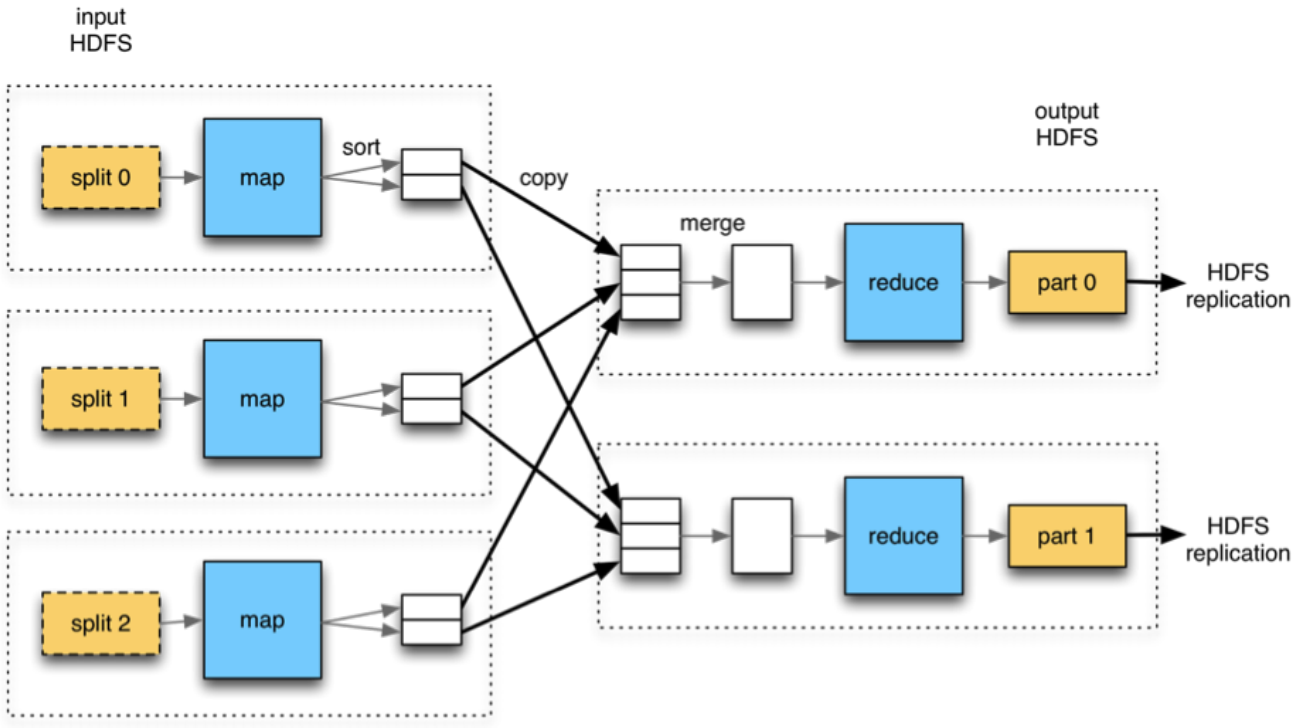
| | | |
|---|---|---|
| 1 CPU cycle | 0.3 ns | 1 s |
| Level 1 cache access | 0.9 ns | 3 s |
| Level 2 cache access | 2.8 ns | 9 s |
| Level 3 cache access | 12.9 ns | 43 s |
| Main memory access | 120 ns | 6 min |
| Solid-state disk I/O | 50-150 $\mu$s | 2-6 days |
| Rotational disk I/O | 1-10 ms | 1-12 months |
| Internet: SF to NYC | 40 ms | 4 years |
| Internet: SF to UK | 81 ms | 8 years |
| Internet: SF to Australia | 183 ms | 19 years |
| OS virtualization reboot | 4 s | 423 years |
| SCSI command time-out | 30 s | 3000 years |
| Hardware virtualization reboot | 40 s | 4000 years |
| Physical system reboot | 5 m | 32 millenia |



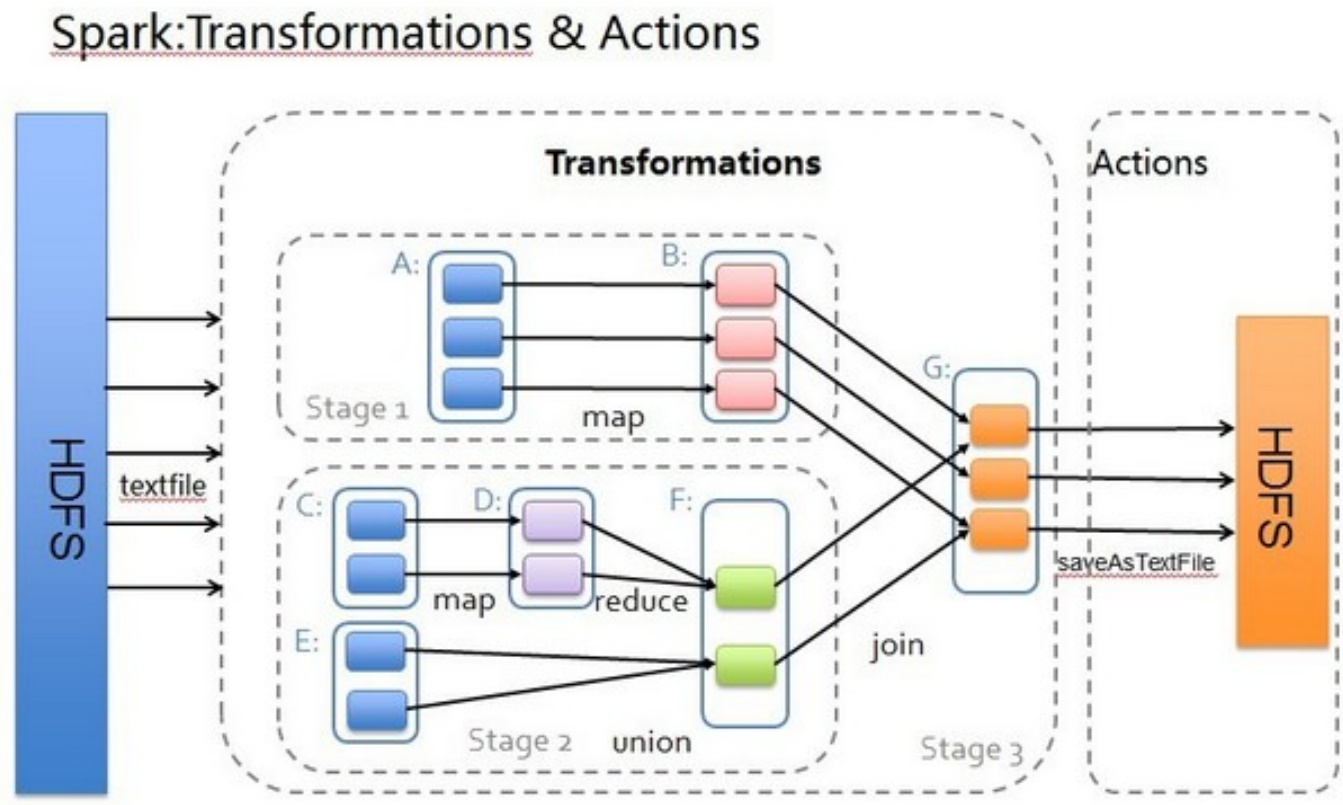

- Distributed (a/b)
- Parallel (c)



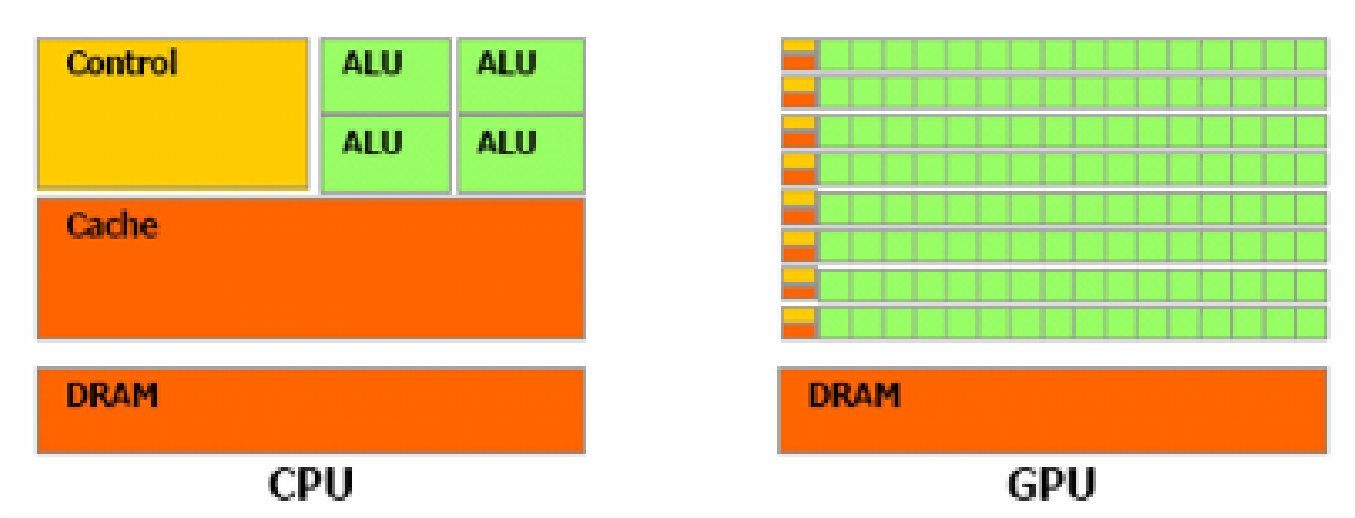

- Several processors/cores with the same shared ram.
- No too expensive transfer between core.
- Strategies:
  - Independent batch
  - Parallelization technique limiting information transfer...
- System memory limitation!




- Data transfer through disk and networked file system!
- Hadoop: Node failure handling and ecosystem.




- Strategy: keep everything as much as possible in memory...




- Combine different processor types...
- CPU $<$ DSP $<$ FPGA $<$ ASICS

# Data Science Challenges

## Outline

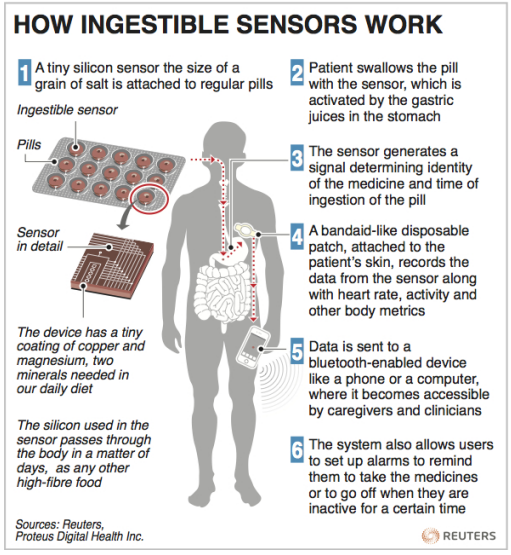
- Applied math **AND** Computer science
- Huge importance of domain specific knowledge: physics, signal processing, biology, health, marketing...

**Some joint math/computer science challenges**

- Data acquisition
- Unstructured data and their representation
- Huge dataset and computation
- High dimensional data and model selection
- Learning with less supervision
- Visualization

# Data Science Challenges

## Data acquisition

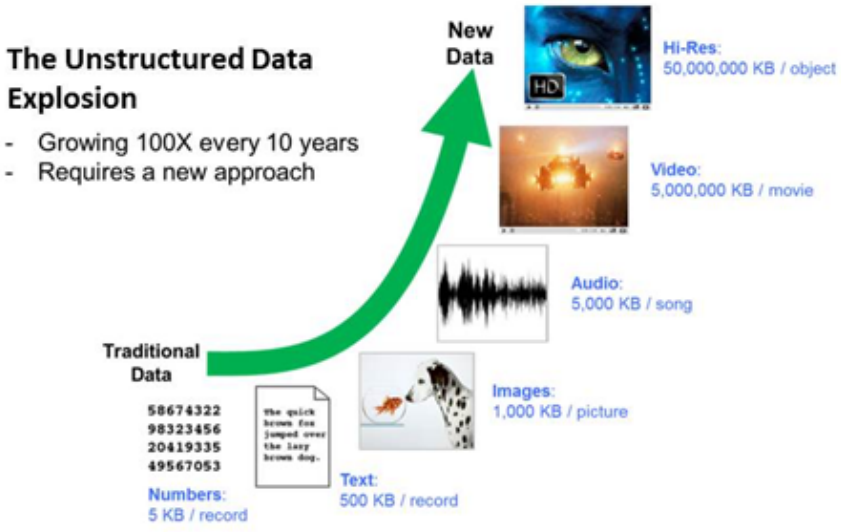


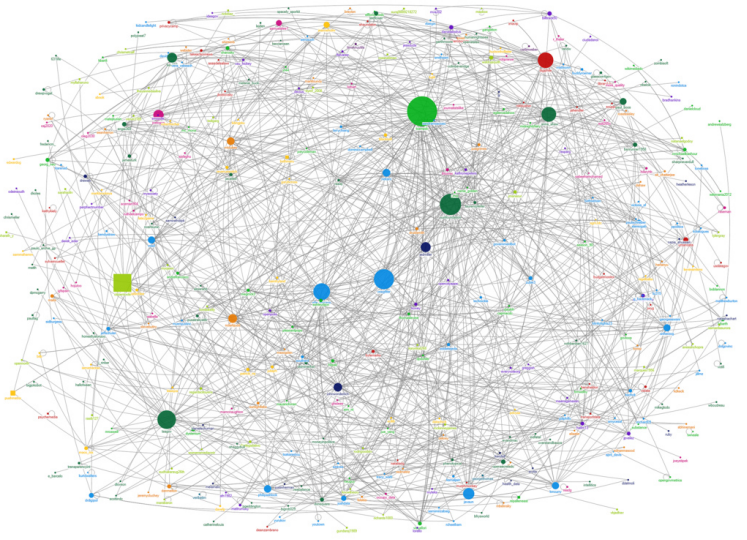

### Some challenges

- How to measure new things?
- How to choose what to measure?
- How to deal with distributed sensors?
- How to look for new sources of informations?

# Data Science Challenges

## Unstructured Data



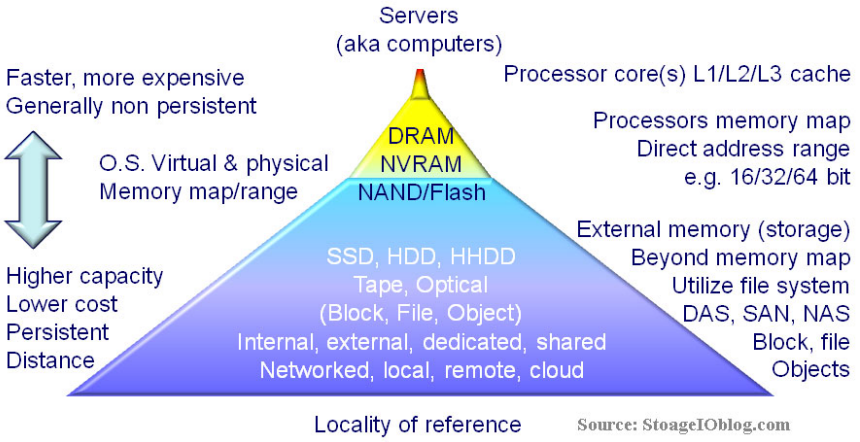



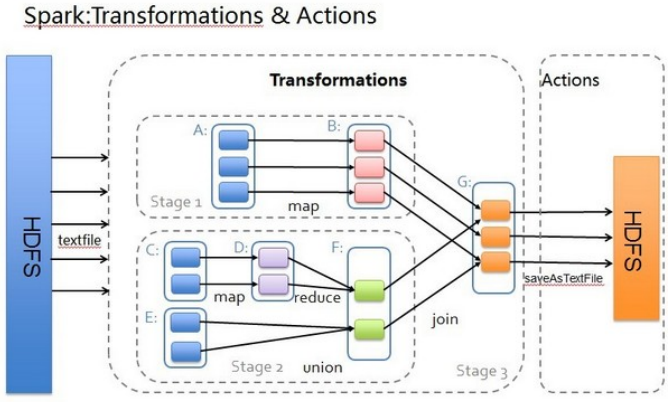
### Some challenges

- How to store efficiently the data?
- How to describe (model) them to be able to process them?
- How to combine data of different nature?
- How to learn dynamics?

# Data Science Challenges

## Huge Dataset

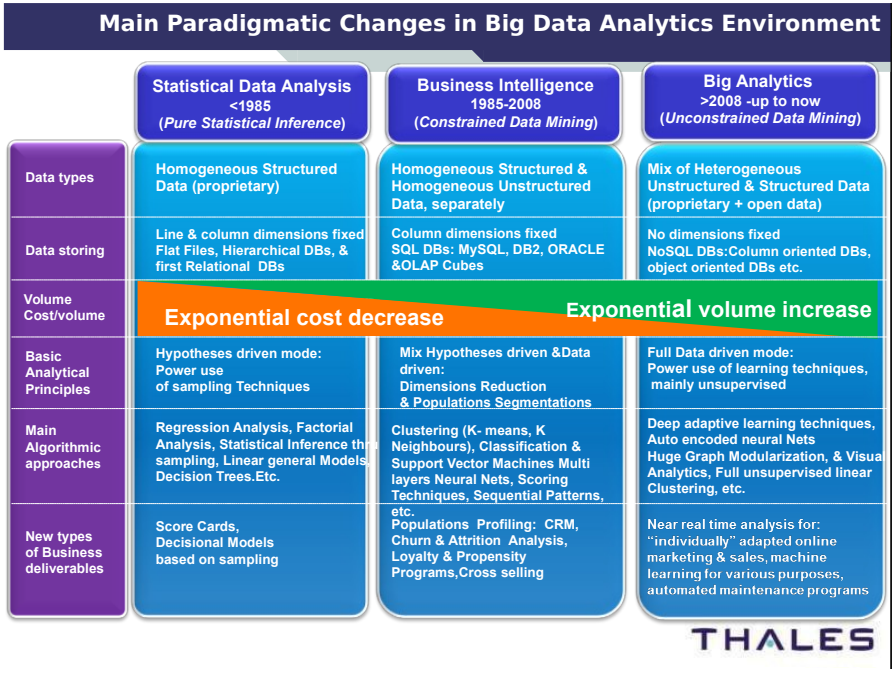



## Some challenges

- How to take into account the locality of the data?
- How to construct distributed architectures?
- How to design adapted algorithms?

# Data Science Challenges
## High Dimensional Data

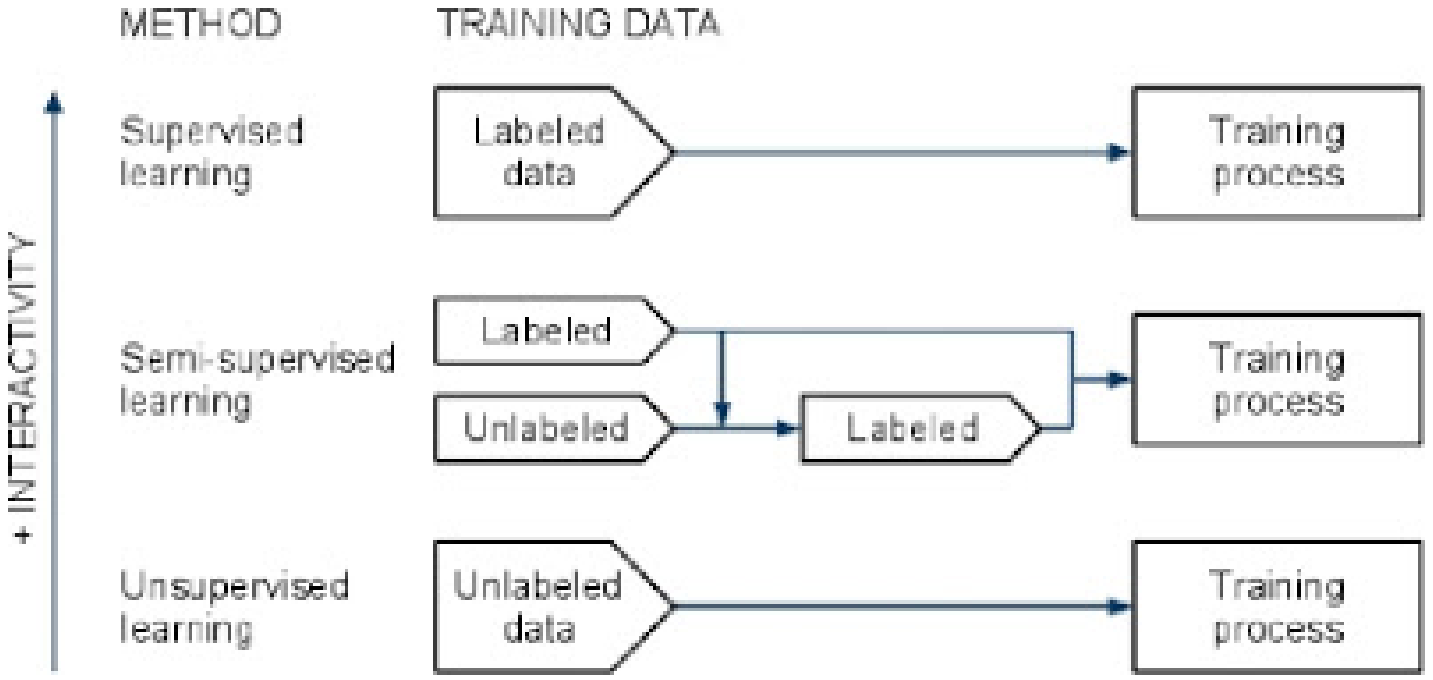



## Some challenges

- How to describe (model) the data?
- How to reduce the data dimensionality?
- How to select/mix models?

# Data Science Challenges

## Learning and Supervision

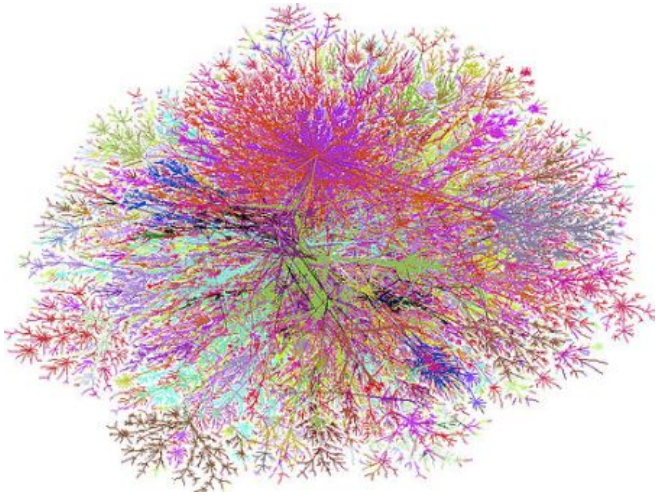


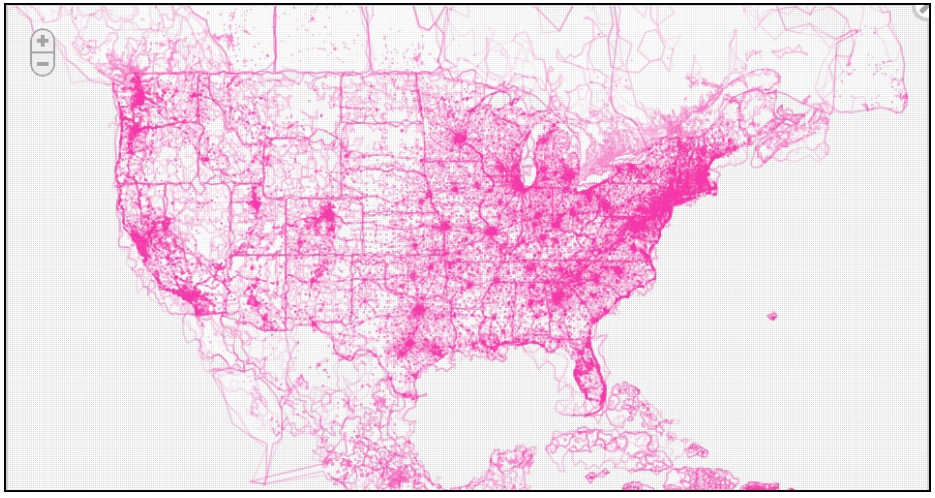

### Some challenges

- How to learn with the less possible interactions?
- How to learn simultaneously several related tasks?

# Data Science Challenges

## Visualization





### Some challenges

- How to look at the data?
- How to present results?
- How to help taking better informed decision?

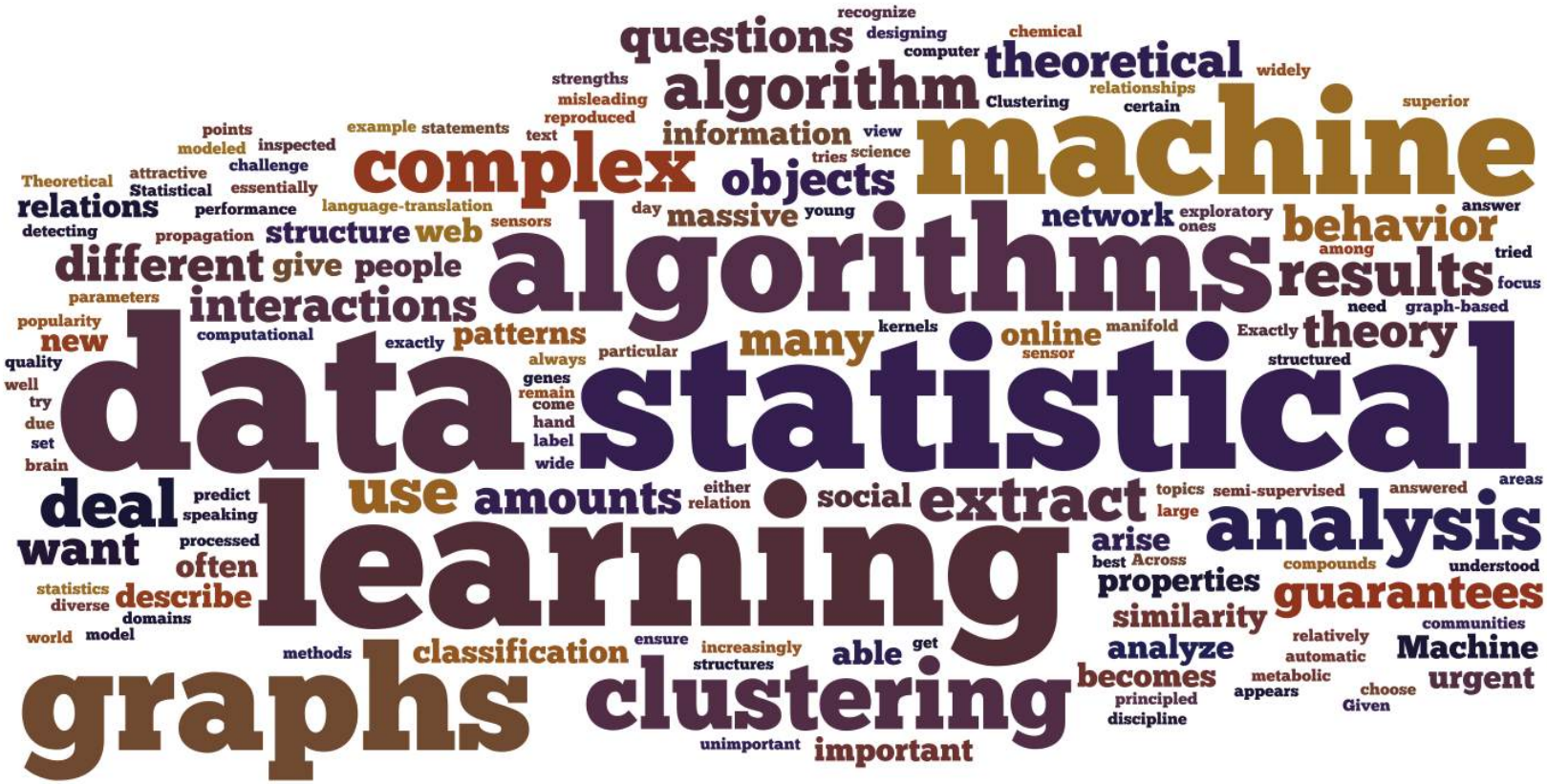
# Vocabulary of Data Science
## Outline

# Vocabulary of Data Science

## Lots of words

# Vocabulary of Data Science

## Lots of words



**Data mining.** Extract patterns from data by combining methods from statistics, machine learning and data processing technologies. *Example*: market basket analysis to model the purchase behavior of customers.

**Machine learning.** Design and develop algorithms allowing computers to learn from data, in order to take intelligent decisions automatically. *Example*: Natural language processing.

**Statistics.** Collection, organization, and interpretation of data. Mathematical methods to construct quantitative assessments of errors and risks when taking decisions, estimating parameters and doing predictions. *Example*: quantitative assessments of relationships between variables, computing confidence intervals for model parameters, hypothesis testing.

**Natural language processing (NLP).** Specialization of machine learning and linguistics that builds algorithms to analyze human (natural) language. *Example*: sentiment analysis on social networks.

**Network analysis.** Characterize relationships among nodes in a graph or a network, understand the communities, the influence of nodes on the others, understand how information travels in the network. *Example*: identify key opinion leaders in a social network, identify the information flows in a large company

**Predictive modeling.** Use of a mathematical model to predict an outcome, e.g. regression, classification, etc. *Example.* Predict the probability that a customer will churn.

**Supervised learning.** Machine learning techniques that infer a function or a relationship from a set of training data. *Examples*: classification, regression

**Unsupervised learning.** Machine learning techniques that finds structure in unlabeled data. *Example*: clustering is a part of unsupervised learning

**Visualization.** Techniques used for representation of data by creating images, diagrams, animations, in order to communicate, understand, explore and improve understanding of data.

**Labels.** Characteristics / categories of interest in points of data. This is the information one wants to predict in supervised learning.

**Features.** A set of information about a point of data (a customer, a company, a country, etc.)

**Clustering.** Algorithms used in unsupervised learning to assign a group to each data point. Groups are called clusters. *Example*: customer segmentation in a e-commerce platform

**Classification.** Algorithms used in supervised learning to predict the labels of data points. It relies on the training of a model or algorithm using labeled data. *Keywords*: Logistic Regression, SVM, CART, Boosting, etc.

**Feature selection**. Algorithms in machine learning that select features that best explain an outcome. *Example*. In biology, find the genetic informations that best explains a patient's response to a drug.

**Generalization.** Ability of a predictive algorithm to generalize: give good predictive results on a sample different than to one used to train the algorithm.

**Parameters.** A set of coefficients (vectors, matrices), that specifies a model. *Example*: the mean and standard deviation of a Gaussian distribution

**Statistical model.** A mathematical formulation that attempts to explain how data is generated. *Example*: data is generated by a multivariate Gaussian distribution

**Likelihood.** The probability that data is generated by a model for some parameters choice

**Optimization.** Study and design of numerical algorithms used to (but not only) minimize or maximize functions. *Example*: optimization of a likelihood is called the **training** step in machine learning.

**Goodness-of-fit.** A quantity that assesses the closeness of a (trained) model to data. *Example*. The least-squares error for linear regression.

**Over-fitting**. Something that must be avoided to have a good predictive performance on new data.

**Cross-validation**. Splitting of data into several subsets. A model is trained on a subset and tested on another, to check its goodness-of-fit both on data used for training, but on new data as well.

**Structured data**. Data structured in fixed fields. *Example*: relational databases or excel spreadsheets.

**Semi-structured data.** Data not structured in fixed fields but contain markers to separate data elements. *Example*: XML or HTML-tagged text.

**Unstructured data.** Data not structured in fixed fields. *Example*: books, articles, body of e-mail messages, audio, image and video data, etc.

**Metadata.** Data that describes the content and context of data: creation, purpose, time and date, author, etc.

**Data fusion and data integration.** Set of techniques that integrate and analyze data from multiple sources, instead of analyzing single sources of data. *Example*: combine analysis of social network data with NLP and real-time sales data, to assess the effect of a marketing campaign on customer sentiment and purchases

**Cloud computing.** A computing paradigm where computing resources are configured as a distributed system, which provides a service through a network.

**Distributed system.** Several computers, communicating through a network, used to solve a common storing or computational problem. Aim is higher performance at a lower cost, higher reliability and scalability.

**Relational database.** A database consisting of collections of tables (relations), namely data are stored in rows and columns. SQL is the most widely used language for managing relational databases.

**Non-relational database.** A database that does not store data in tables (rows and columns).

**SQL.** Acronym for Structured Query Language. It is a computer language designed for managing data in relational databases. *Example*. insert, query, update, delete data, manage database structures, and control access to data in the database.

**NoSQL.** A group of database management systems. Data is not stored in tables like in Relational database. It does not rely on the mathematical relationship between tables. It gives a way of storing and retrieving unstructured data quickly.

**Hbase.** A distributed, non-relational database. It is managed as a project of the Apache Software foundation and a part of Hadoop.

**Hadoop.** A framework that supports large scale data processing by allowing the decomposition of large tasks into smaller tasks, that are executed in parallel, on independently slices of the data and then finally merged to answer to the task.

**MapReduce.** A software framework introduced by Google for processing huge datasets on a distributed system. Implemented in Hadoop. It supports large scale data processing by decomposing large tasks into smaller tasks, executed in parallel, on independent parts of data and finally merged to answer to the task.

**Stream processing.** Technologies designed to process large real-time streams of event data. *Example*: high-frequency algorithmic trading, analysis of Twitter data streams

# Bibliography

## Bibliography


T. Hastie, R. Tibshirani, and J. Friedman (2009)
The Elements of Statistical Learning
*Springer Series in Statistics.*

G. James, D. Witten, T. Hastie and R. Tibshirani (2013)
An Introduction to Statistical Learning with Applications in R
*Springer Series in Statistics.*

B. Schölkopf, A. Smola (2002)
Learning with kernels.
*The MIT Press*

R. Schutt, and C. O'Neil (2014)
Doing Data Science: Straight talk from the frontline
*O'Reilly*