Big data computing

Instructors: Prof. Luca Becchetti, Prof. Ioannis Chatzigiannakis

Goals

- Advanced tools used in data analysis that are currently not covered in the basic curriculum
  - Algorithms and intuitions/theory behind
  - Implementation in realistic scenarios
- Emphasis is on massive data sets or “big data”
- More about meaning of “big” later in this lecture

Prerequisites

- Linear algebra
- Calculus and basic knowledge of probability theory and statistics
- Programming, fundamental algorithms and data structures
- Attendance at the data mining class is a plus
- Languages
  - We mostly use Python 3.x
  - Other languages are welcome, but not all of them can be used on the distributed platforms we consider
  - We give you hints, reviewing/learning the basics of these language (if needed) is up to you

Course details

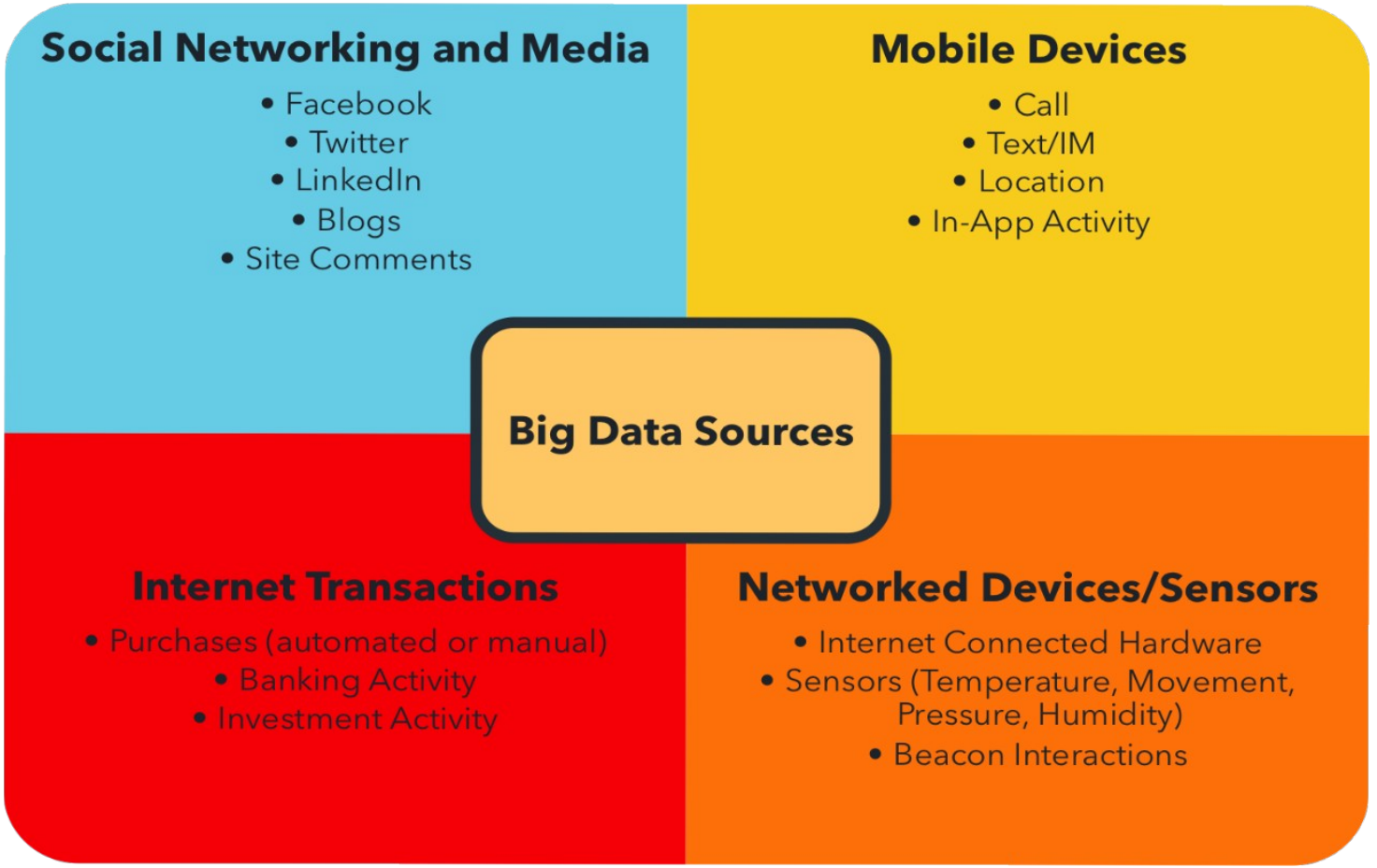
- **Where**: via Ariosto 25
- **When**:
  - Mondays, 11 – 12, room A7
  - Thursdays 4pm – 6pm, room B2
  - Fridays, 2pm – 4pm, room A3
- **Web site**: https://piazza.com/uniroma1.it/spring2017/1044406/home

Teaching staff

- Luca Becchetti
  - becchetti@dis.uniroma1.it
  - Office hrs (might change):
    - Wednesdays, 10.30 – 12
    - Thursdays, 11 - 12
- Ioannis Chatzigiannakis
  ichatz@dis.uniroma1.it

Where do data come from?

Data sources



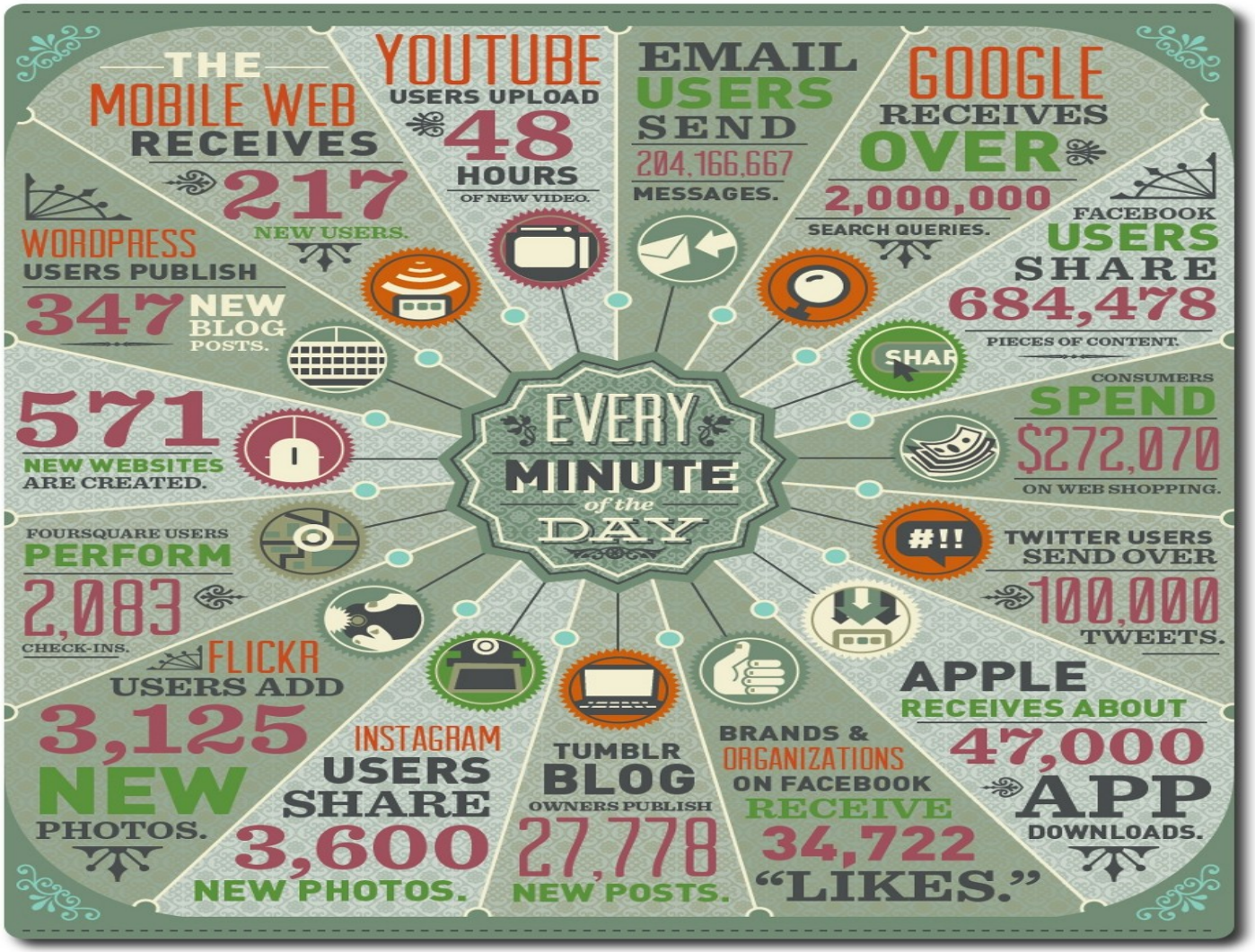
© IBM (IBM's Big data hub)

Digital universe and digital footprint

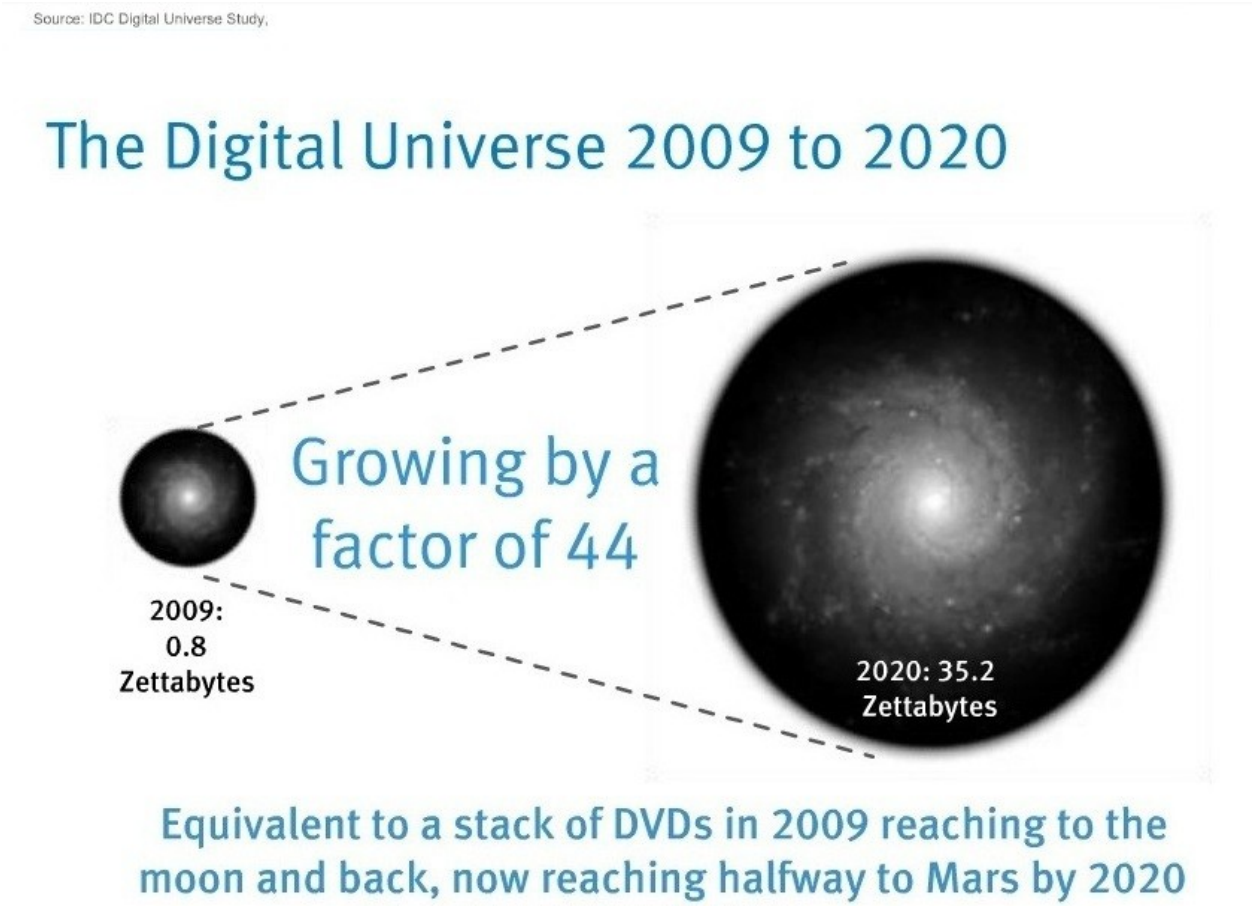
Your digital footprint (or shadow)

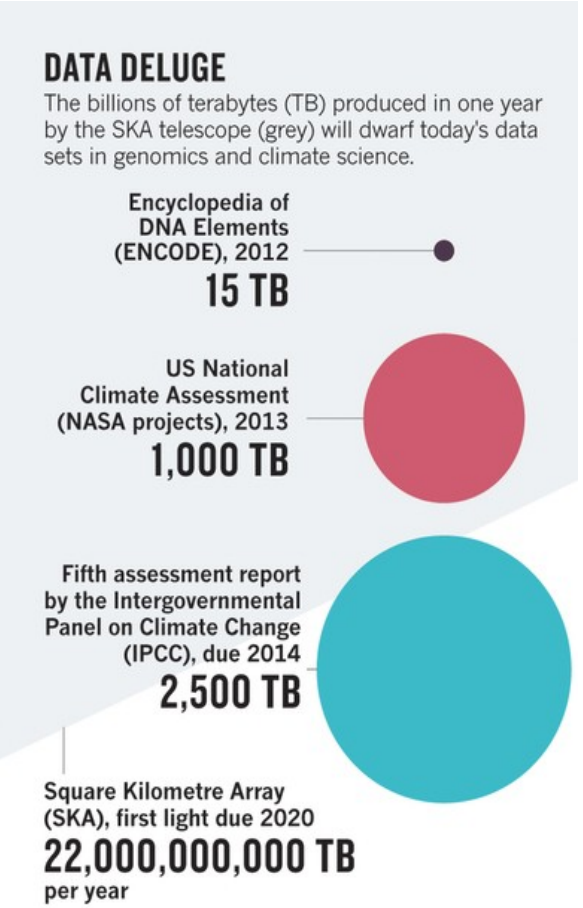
How big is “big”?

Taking a snapshot



The growth of things





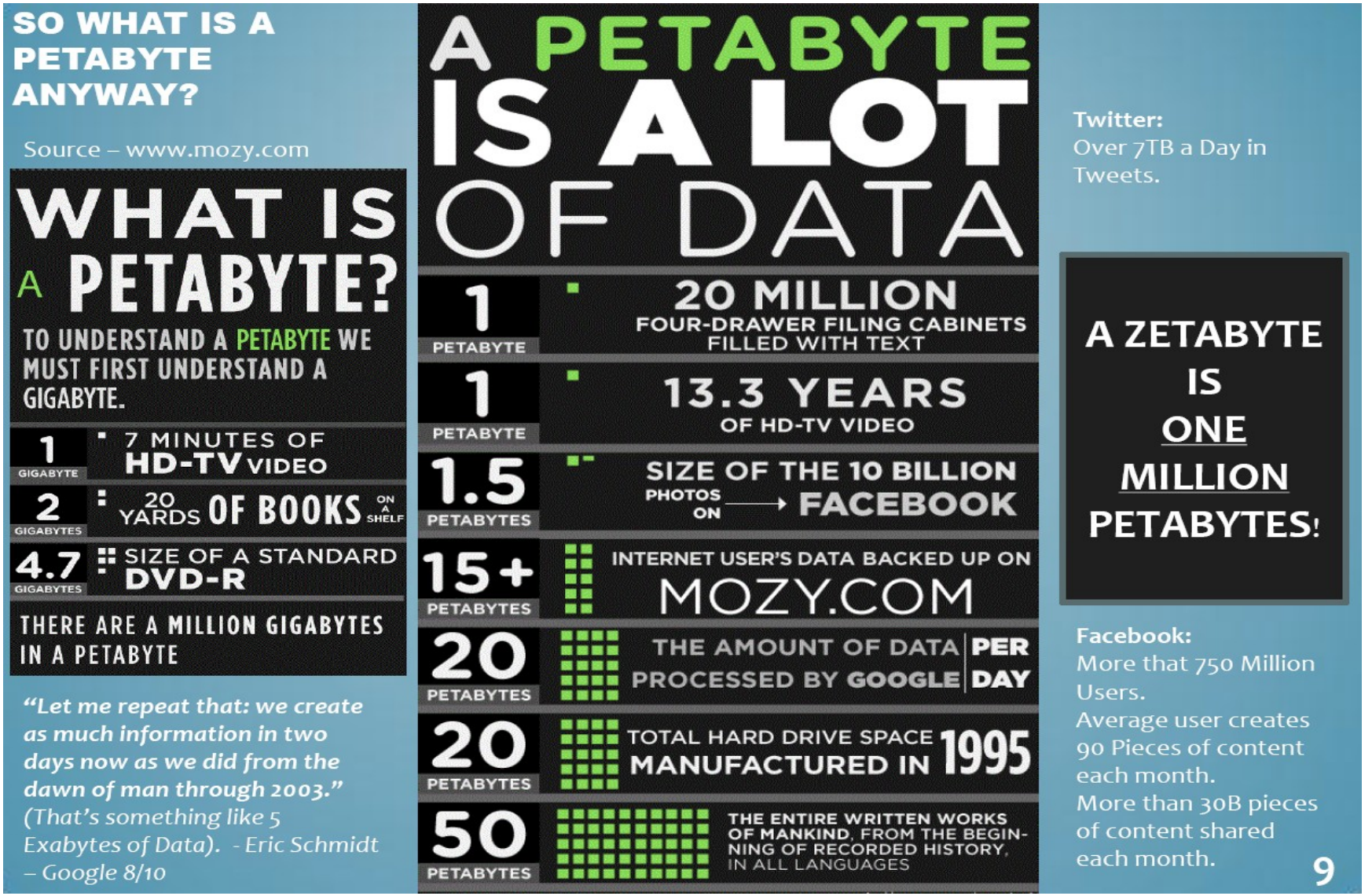
Source: IDC Digital Universe Study

The magnitude of things

Metric prefixes						
Prefix	**Symbol**	**1000^m**	**10^n**	**Decimal**	**English word**[n 1]	**Since**[n 2]
yotta	Y	1000^8	10^{24}	1 000 000 000 000 000 000 000 000	septillion	1991
zetta	Z	1000^7	10^{21}	1 000 000 000 000 000 000 000	sextillion	1991
exa	E	1000^6	10^{18}	1 000 000 000 000 000 000	quintillion	1975
peta	P	1000^5	10^{15}	1 000 000 000 000 000	quadrillion	1975
tera	T	1000^4	10^{12}	1 000 000 000 000	trillion	1960
giga	G	1000^3	10^9	1 000 000 000	billion	1960
mega	M	1000^2	10^6	1 000 000	million	1960

Courtesy: Camil Demetrescu and Irene Finocchi

Example: petabyte



Big data in a nutshell: the 3 V's

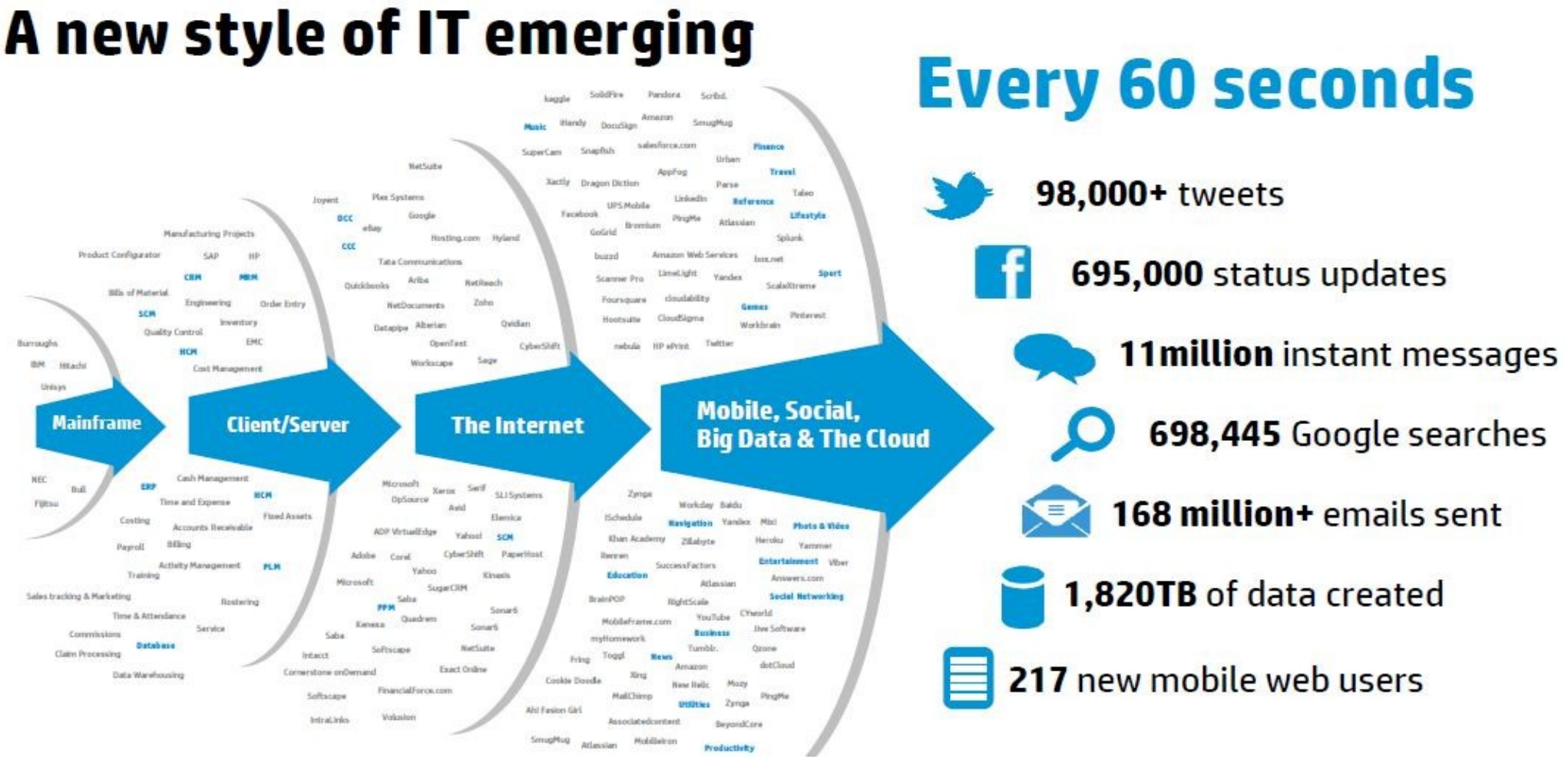
- *“**Big data** is high-**volume**, high-**velocity** and/or high-**variety** information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation.”*
- Source: Gartner at http://www.gartner.com/it-glossary/big-data

Challenges

Volume

- Your data do not fit in main memory
  - A search engine query-log
  - A snapshot of the Web
  - The user pairwise-similarity matrix in a collaborative filtering system
  - ...
- Your data do not fit on secondary storage on a single machine
  - A Web corpus
  - Large Hadron Collider's data (~25 GB/s)
    - http://home.web.cern.ch/about/computing

Velocity



Related: continuous data streams that you cannot permanently store for off-line analysis

Variety

- Heterogeneous sources
- Structured vs unstructured (often unstructured or semi-structured)
- Part of this data is transient (e.g., streamed videos)
- Only part of it is available for analysis

Issues: a random list

- Data may not be stored permanently
- Moving data is expensive
- It may be impossible to keep all data in a single place
- In general, space becomes an issue
- Distributed/parallel programming
- Heterogeneous programming environments
- Performance is main issue
  - A normally trivial computation may prove infeasible
    - e.g., assume you have the feature vectors for 1 million users and you want to compute pairwise similarities. Assume computing similarity between 2 users requires 10 ns (we assume computation is performed in main memory). How long does your computation require using the naïve method?

A look at the data

- Text (possibly with some structure)
  - e.g., Web pages
- Graphs
- Sequences
  - e.g., search engines or recommender systems' logs
- Data often stored as simple text files with more or less standard formats (e.g., csv)

```
2::1207::4::978298478
2::1968::2::978298881
2::3678::3::978299250
2::1244::3::978299143
2::356::5::978299686
2::1245::2::978299200
2::1246::5::978299418
2::3893::1::978299535
2::1247::5::978298652
3::3421::4::978298147
3::1641::2::978298430
3::648::3::978297867
3::1394::4::978298147
3::3534::3::978297068
```

Wrapping it up

Main application goals

- Retrieving items related to a given query (query may be implicit)
  - Search engines
  - Continuous, DB-like queries
    - E.g.: top-k most interesting trends
    - Number of distinct IP flows
  - Classification
    - E.g., which Web pages are spam?
- Recommending items
  - Products (e.g., Amazon)
  - Documents of potential interest (e.g., Google alerts)
  - New links (e.g., new friendships on FB)
- Detecting group of items that are related in some way
  - e.g., communities in social networks, similar Web pages

Seems little but actually a lot! Many applications fit one of the above high level descriptions, which are themselves related

The big picture

Tasks

- Similarity estimation
- Classification and clustering
- Frequent item(sets)
- Link analysis and ranking
- Graph mining

Goals

Retrieving items related to a given query

Recommending items

Identifying groups of similar/related items

Tools/techniques

Hashing | Streaming | Dimensionality reduction | Parallelism

Big data – caveats and traps

When is “big” really big?

- It is a result of your data size and goals vs your computational resources
  - Data volume vs memory
  - Computational/communication complexity vs available resources
- A moving target
  - What is big now may not be in the future

Data collection traps

- Data are the result of a sampling process → potential bias
- Filtering: some attributes may be omitted → loss of possibly important information, introduction of artificial correlations/removal of real ones
- At least part of context is missing

Data analysis traps: Bonferroni's principle

- Data can always present some unusual patterns that look significant but are not
- e.g., a DB table where rows are people and columns are attributes. You may have thousands of attributes and thus make hypotheses that seem to be supported by data only by chance → we'll see some toy examples

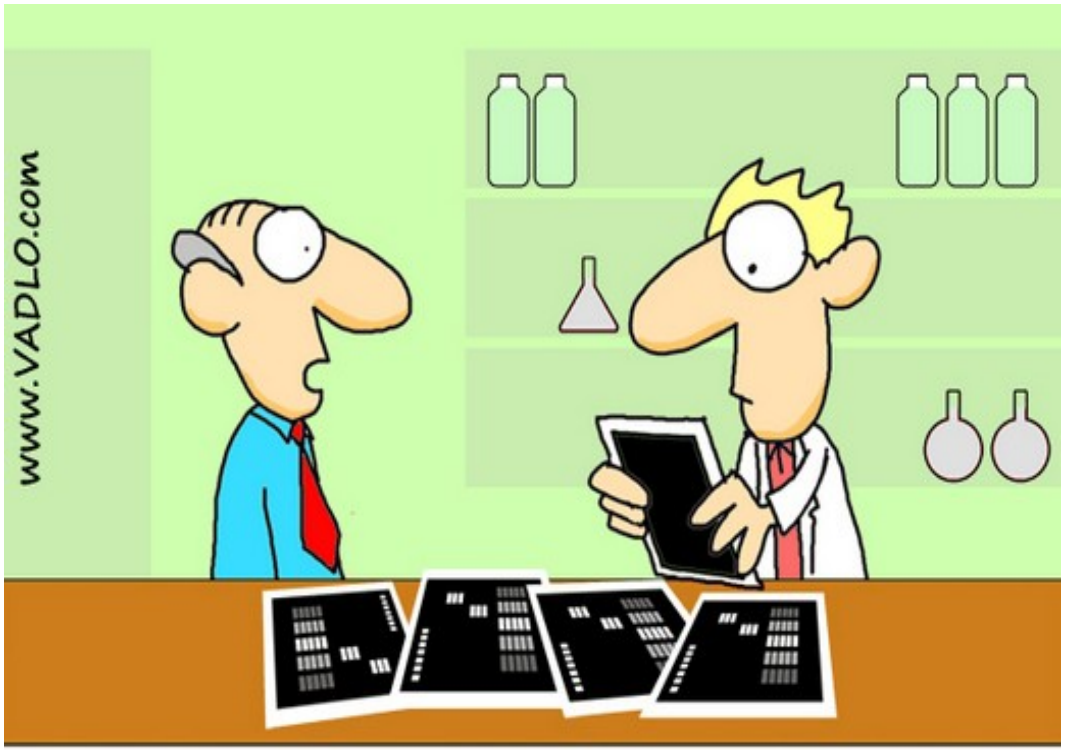
"...if you just grab a few of them [hypotheses] ... because the data seem to suggest that—there's some chance you will get lucky. The data will provide some support. But unless you're actually doing the full-scale engineering statistical analysis to provide some error bars and quantify the errors, it's gambling." IEEE Spectrum's interview with Michael Jordan

Interpretations traps

- Causality vs correlation: does A → B or are they simply correlated?
- Generality: Is the population the data refer to representative of a larger population?
  - We are mostly working on biased samples
  - Information-rich attributes may have been filtered out during data cleaning
  - A lot of context is missing
  - e.g.: using social networking platform data to infer trends in society may be misleading if not done carefully

Wrapping it up - caveats

- Need a lot of knowledge and critical thinking to analyze data
  - Data are “projections” of often complex processes
  - Part of the context is lost in collection or is simply unavailable for collection
  - Maybe a wealth of information in the data, but need tools to tell statistically significant patterns from bogus



“Data don’t make any sense,
we will have to resort to statistics.”