FIT5145 Introduction to Data Science

Module 3

Data Types and Storage

2019 Lecture 6

Monash University

Discussion: Unix Shell

Useful for managing and manipulating **large files**

- **without ever loading them fully into memory**
- using pipes allow us to process files as a stream
- allows us to deal with files that are too big for applications and/or don’t fit into memory

Shell contains many useful commands, like

- less to view large files
- grep to search large files
- awk to process them one line at a time (and cut them down to size for visualising)

Discussion: Factors that Influence Data Science

over and above general growth of hardware

Can you name some?

- business needs
- data analysis and general wrangling tools
- the internet (related to new “computing class”)
- big business recognition

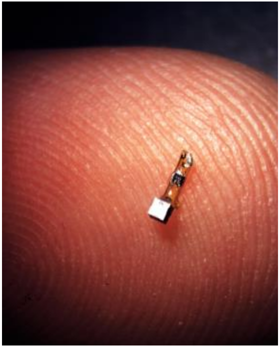
FLUX Question
New Classes of Computing

Remember Bell's law ... new classes of computing every decade.

Can you suggest some new classes of computing?

Discussion:
New Classes of Computing

mind-reading or mind-control devices



in-body devices

NB. sounds like science fiction but we know R&D exists in all these areas!

Unit Schedule: Modules

Module	Week	Content
1.	1	Overview and look at projects
	2	(Job) roles, and the impact
2.	3	Data business models / application areas
3.	4	**Characterising data and "big" data**
	5	**Data sources and case studies**
4.	6	Resources and standards
	7	Resources case studies
5.	8	Data analysis theory
	9	Regression and decision trees
	10	Data analysis process
6.	11	Issues in data management
	12	GUEST SPEAKER & EXAM INFO

Learning Outcomes (Week 6)

By the end of this week you should be able to:

- Characterize different database types
- Differentiate between SQL and NoSQL databases
- Define what distributed processing is
- Analyse the Map-Reduce framework
- Differentiate between Hadoop and Spark
- Apply R/shell commands to read/manipulate big data files

Big Data Processing
(ePub section 3.4)

processing data at scale, especially for analysis

- databases

  storing and accessing data

- distributed processing

  breaking up computation to scale it up

Business Context

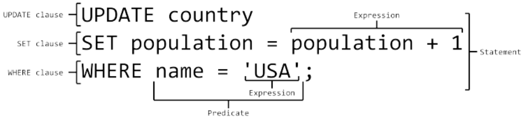
- businesses function in a continuously changing environment:
  - fixed formats as per RDBMS not suitable
- businesses function in a continuously changing environment:
  - usage varies, requires complex analytical queries
- need to reach insights faster and act on them in real time
  - stream processing

Big Data Processing:
Databases

storing and accessing data

SQL Review

- Relational Database Management Systems (RDBMS)
- SQL ::= structured query language



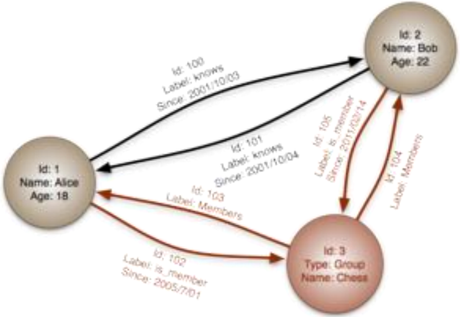
- rather like large scale set of Excel spreadsheets with better indexing and retrieval
- transaction oriented with support for correctness, distribution, ...

JSON Example

```json
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    }
  ],
  "children": [],
  "spouse": null
}
```

- no fixed format
- semi-structured, key-value pairs, hierarchical
- “friendly” alternative to XML
- self-documenting structure

Graph Database Example



- stores graph, commonly as triples, subject, verb, object
- commonly used to store Linked Open Data

Database Background Concepts

in-database analytics: the analytics is done within the DB

in-memory database: the DB content resides memory

cache: data stored in-memory

key-value: *value* accessible by *key*, *e.g.*, hash table

information silo: an insular information system incapable of reciprocal operation with other, related information systems

- if two big banks merge, then initially their RDBMSs will be siloed
- in a big insurance company, auto and home insurance customer RDBMSs may be siloed

Database Background Concepts

Many NoSQL and SQL DBs offer:

- large scale, distributed processing
- robustness achieved
- general query languages
- some notion of consistency

  e.g. “eventually” as nodes spread updates

Beyond SQL Databases

Type	Notes
RDBMS	SQL
Object DB	navigate network
Doc. DB	JSON like, Javascript like queries
key-val cache	in-memory
key-val store	not in-memory but highly optimised
tabular key-val	relational-like, “wide column store”
graph DB	RDF, SPARQL,

SQL and Beyond SQL Databases (NoSQL)

- Use SQL database when:
  - data is structured and unchanging
- Use NoSQL database when:
  - Storing large volume of data with little to no structure
  - Data changes rapidly
- NoSQL databases offer a rich variety beyond traditional relational.

Overview: Databases

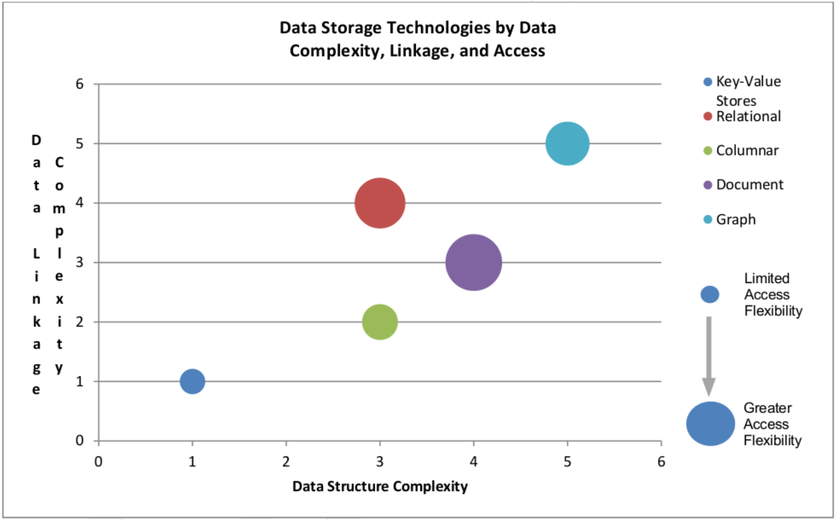


Figure 4: Data Storage Technologies

Big Data Processing:
Distributed processing

breaking up computation to scale it up

Overview: Processing

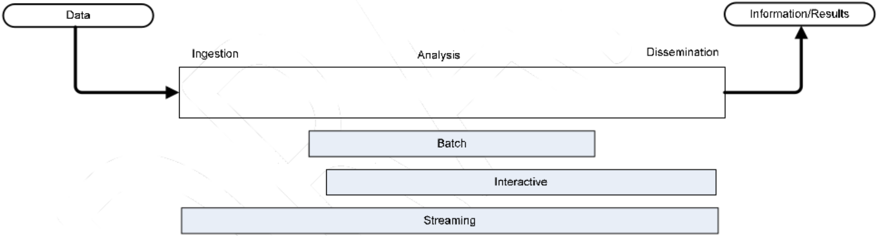


Figure 5: Information Flow

Interactive: bringing humans into the loop

Streaming: massive data streaming through system with little storage

Batch: data stored and analysed in large blocks, “batches,” easier to develop and analyse

Processing Background Concepts

in-memory: in RAM, *i.e.*, not going to disk

parallel processing: performing tasks in parallel

distributed computing: across multiple machines

scalability: to handle a growing amount of work; to be enlarged to accommodate growth (not just "big")

data parallel: processing can be done independently on separate chunks of data

- yes: process all documents in a collection to extract names
- no: convert a wiring diagram into a physical design (optimisation)

FLUX Question

Which one of the following tasks is not easy to make data parallel?

A. Face recognition in 1M images
B. Invert a large matrix
C. Looking for common 3-4 word phrases in a collection of documents

Distributed Analytics

- legacy systems provide powerful statistical tools on the desktop
    - SAS, R, Matlab

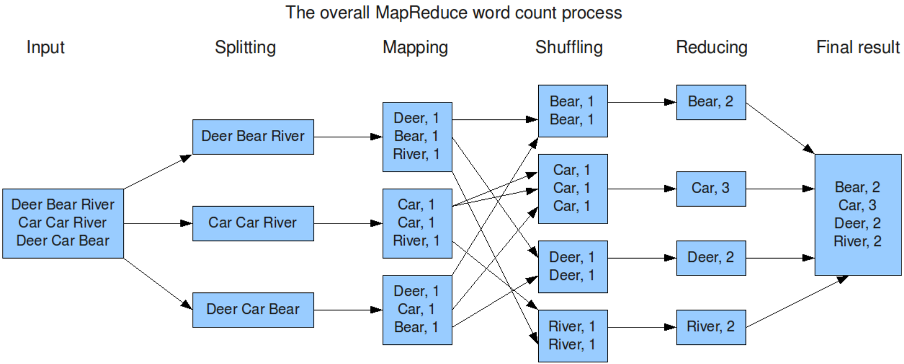
  but often-times without distributed or multi-processor support
- supporting distributed/multi-processor computation requires special redesign of algorithms

Map-Reduce

Simple distributed processing framework developed at Google

- published by Dean and Ghemawat of Google in 2004
- **intended to run on commodity hardware**; so has fault-tolerant infrastructure
- from a distributed systems perspective, is quite simple

Map-Reduce Example



for a simple word-count task: (1) divide data across machines (2) `map()` to key-value pairs (3) sort and `merge()` identical keys

Map-Reduce, cont.

- requires simple data parallelism followed by some merge ("reduce") process
- stopped using by Google probably in 2005
- Google now uses *"Cloud Dataflow"* (and *here*), available commercially, as open source

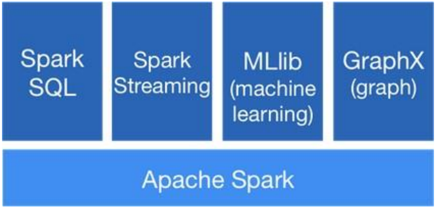
Hadoop

Open-source Java implementation of Map-Reduce

- originally developed by *Doug Cutting* while at Yahoo!
- architecture:
  - Common: Java libraries and utilities
  - MapReduce: core paradigm
- huge tool ecosystem
- well passed the peak of the hype curve

Spark

- another (open source) Apache top-level project at *Apache Spark*
- developed at *AMPLab* at UC Berkeley
- builds on Hadoop infrastructure
- interfaces in Java, Scala, Python, R
- provides in-memory analytics
- works with some of the Hadoop ecosystem



FLUX Question

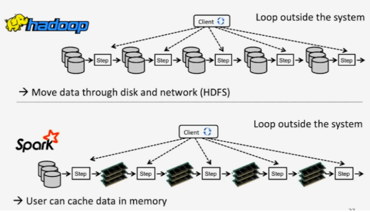
Which one of the following is suitable for real-time data processing?

A. Hadoop
B. Spark

Summary: Hadoop and Spark

- Hadoop provides an inexpensive and open source platform for parallelising processing:
  - based on a simple Map-Reduce architecture
  - not suited to streaming (suitable for offline processing)

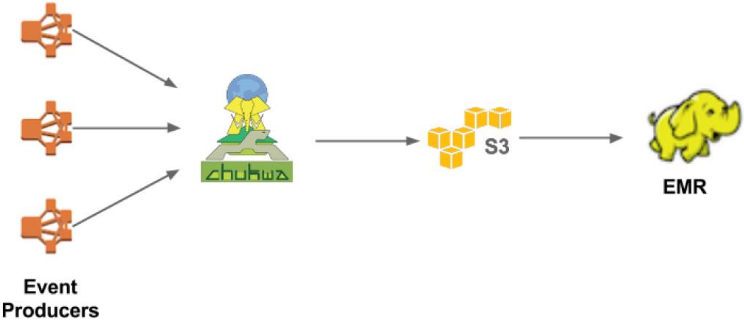
- Spark is a more recent development than Hadoop
  - includes Map-Reduce capabilities
  - provides real-time, in-memory processing
  - much faster than Hadoop



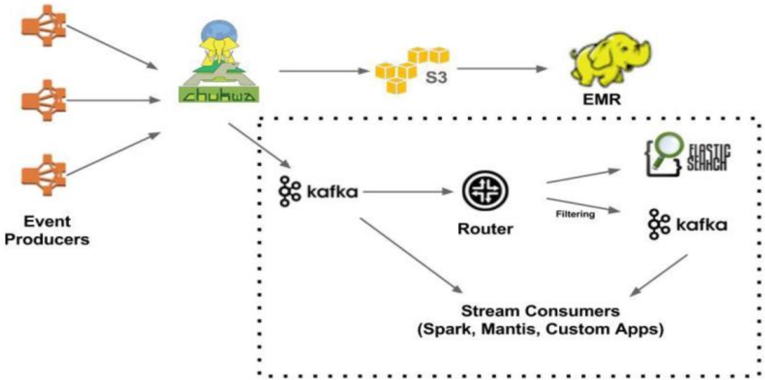
Evolution of the Netflix Data Pipeline

- Here are some statistics about Netflix data pipeline:
    - ~500 billion events and ~1.3 PB per day
    - ~8 million events and ~24 GB per second during peak hours
- There are several hundred event streams flowing through the pipeline. For example:
    - Video viewing activities
    - UI activities
    - Error logs
    - Performance events
    - Troubleshooting & diagnostic events

Netflix Data Pipeline
V1.0 Chukwa pipeline

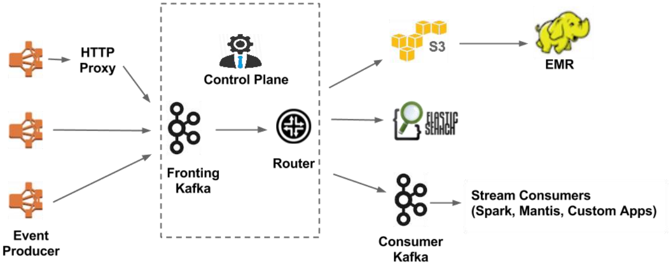


Netflix Data Pipeline: V1.5 Chukwa pipeline with real-time branch



Netflix Data Stack

Simplified view using Apache Kafka, Elastic Search, AWS S3, Apache Spark, Apache Hadoop, and EMR.



see *[Architecture of Giants: Data Stacks]()*

The Machine Learning Renaissance

Mike Olson (co-founded Cloudera in 2008) says without big data and a platform to manage big data, machine learning and artificial intelligence just don't work.

See *the machine learning renaissance* starting at 60 seconds.

Data Case Studies
(ePub section 3.3)

examples of different kinds of data

- illustrating the process
  - a quick walkthrough illustrating the steps

NIST Case Studies

they give us a catalogue of examples and an infrastructure for doing our analysis

Reminder: NIST Analysis

data sources: where the data comes from

data volume: how much there is

data velocity: how it changes over time

data variety: what different kinds of data there is

data veracity: correctness problems in the data

software: software needed to do the work

analytics: broadly, what sorts of statistical analysis and visualisation needed

processing: broadly, computational requirements

capabilities: broadly, key requirements of the operational system

security/privacy: nature of needs here

lifecycle: ongoing requirements

other: noteable factors

Motivating Examples

not really case studies, but some good motivating examples of whats out there

Case Studies

"Visualizing the world's Twitter data – Jer Thorp", a TEDYouth 2012 Talk, former New York Times data artist-in-residence Jer Thorp (video, 6mins)

National Map (Youtube, 14 mins) is a website for map-based access to Australian spatial data from government agencies. The website is *http://nationalmap.gov.au/*.

"Style Stalking; The Stochastic Patterns that Drive Fashion Trends", by Karen Moon from Strata+Hadoop World 2014 (video, 10 minutes)

Panama Papers, leaked papers (11.5M) on financial transactions, *motivations for using data science*, and *how analysed* (Wired, 2016).

Next: Module 4
Data Resources, Processes, Standards and Tools