Calcul Parallèle (Map Reduce)

Olivier Curé

Université Paris-Est Marne la Vallée , LIGM UMR CNRS 8049, France

November 6, 2016

- Intro to distributed Systems
- MapReduce / Hadoop
- Spark
  - Spark Core
  - SparkSQL and DataFrame
  - GraphX
  - Spark streaming

Important concepts/terms

- Scalability
- Performance
- Latency
- Availability
- Fault tolerance

Scalability

- The ability of a system, network or process, to handle a growing amount of work in a capable manner or its ability to be enlarged to accommodate that growth.
- Meet the needs of users as scale increases.
- Different forms:
  - Size: adding more nodes should make the system linearly faster, growing the dataset should not increase latency.
  - Geographic: add data centers to reduce the time needed to answer user queries.
  - Administrative: adding more nodes should not increases the administrative costs of the complete system.

Latency #1

- The time between the initiation of something and the time when its occurrence has an impact or is visible.
- Example in a database context: how fast a write become available to readers.

Latency #2

Table 2.2 Example Time Scale of System Latencies

Event	Latency	Scaled
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 (DRAM, from CPU)	120 ns	6 min
Solid-state disk I/O (flash memory)	50–150 µs	2–6 days
Rotational disk I/O	1–10 ms	1–12 months
Internet: San Francisco to New York	40 ms	4 years
Internet: San Francisco to United Kingdom	81 ms	8 years
Internet: San Francisco to Australia	183 ms	19 years
TCP packet retransmit	1–3 s	105–317 years
OS virtualization system reboot	4 s	423 years
SCSI command time-out	30 s	3 millennia
Hardware (HW) virtualization system reboot	40 s	4 millennia
Physical system reboot	5 m	32 millennia

Availability

- The proportion of time a system is in a running condition.
- If a user cannot access the system, that system is unavailable.

Fault tolerance

- Most systems fail
- In a distributed system context, if one component fails, is the overall system still working properly?
- Fault tolerance is the ability of a system to behave in a well-defined manner once fault occurs.
- Systems can anticipate faults (monitoring) and/or cope with them.
- A reliable system is one that continues to work correctly, even when things go wrong.

Relationships between concepts

- Increase of independent nodes :
  - increases the probability of failure of the system: reducing availability and increasing administrative tasks.
  - may increase the need for machine communication: reducing performance as the system scales
- Increase in geographic distance increases the minimum latency for machine communication: reducing performance for some operations.

Distributed System

- A distributed system is an application that coordinates the actions of several computers to achieve a specific task.
- This coordination is achieved by exchanging messages (pieces of data) $\rightarrow$ shared-nothing architecture $\rightarrow$ no shared memory, no shared disk.
- The system relies on a network that connects the computers and handles the routing of messages $\rightarrow$ Local area networks (LAN), Peer to peer (P2P) networks
- Client (nodes) and Server (nodes) are communicating software components: we assimilate them with the machines they run on.

LAN

- LAN to connect hundreds or thousands of machines in data centers
  - 3 communication levels: at the rack level, at the router level (between racks), at the cluster level.
  - All with different bandwidth characteristics.
  - A Google data center : 100-200 racks with 40 servers each.
- P2P
  - A kind of overlay network (graph structure build over a native physical network.
  - Ususally the internet: peers communicate with messages sent over the Internet.

Data locality principle

- Consider you need to process 1TB of data.
  - it takes more than 2.5 hours to read with a sequential access from a single machine (disk at 10MB/S)
  - with a parallel access over 100 disks and a single machine, it takes 1.3 minute (read 10GB from each disk). CPU of the computer is overwhelmed.
  - distributed access: 100 disks and 100 machines. Machines are connected by a network (100MB/s to 1GB/s). CPU is not overwhelmed.

	Latency	Bandwidth
LAN	$\approx$ 1-2ms	$\approx$ 1GB/s (single rack); $\approx$ 100MB/s (router)
Disk	$\approx$ 5ms	at most 100MB/s
Internet	10 to 100ms	few MBs

Data locality principle#2

- The previous example is fine for batch processing: sequential read.
- Distribution is less relevant to speed up ops in the case of random reads.
- Data locality principle: process data locally. That is move the programs, not the data.
- Summary:
  - disk transfer is a bottleneck for batch processing
  - disk seek is a bottleneck for transactional processing

"A hundred years ago, companies stopped generating their own power with steam engines and dynamos and plugged into the newly built electric grid. The cheap power pumped out by electric utilities didnt just change how businesses operate. It set off a chain reaction of economic and social transformations that brought the modern world into existence. Today, a similar revolution is under way. Hooked up to the Internets global computing grid, massive information-processing plants have begun pumping data and software code into our homes and businesses. This time, its computing thats turning into a utility."[1]

[1] http://www.nicholasgcarr.com/bigswitch/

- Several aspects of cloud computing: business, market, technical, research, etc.
- "a cloud provides on demand resources and services over the Internet, usually at the scale and with the reliability of a data center"[2]
- Google's perspective on cloud computing:
  - user-centric (owner decides to share apps, docs, etc.)
  - task-centric (focus on what needs to be doneand how)
  - powerful (access to a cluster of machines)
  - accessible (many sources of data on the Web)
  - 'intelligent' (analysis of datasets)
  - programmable (data integrity, data distribution, replication)

[2]Grossman, R. L. and Gu, Y. (2009). On the varieties of clouds for data intensive computing. Q. Bull. IEEE TC on Data Eng., 32(1):4450.

- Used to get computing, storage and networking resources
- It is a natural evolution and combination of different computing models found on the Web. But where is it coming from?
- Architectures through computer science history:
  - Centralized in the 60s: light clients connected to mainframes
  - Decentralized in the 90s: rich clients connecting to database servers
  - Centralized again from 95: client browsing Web servers:
    - Based on HTTP and HTML (invented in early 90s by Tim Berners-Lee)
  - Architecture based on SOA

Antecedents

- **Client Server computing**: centralized storage and processing but limited flexibility, bad performances and user-centric focus
- **Peer to Peer**: every computer is a client and a server $\Rightarrow$ decentralization.
- **Distributed computing**: subset of P2P where idled PCs are used for computation (e.g. SETI@Home)

Cloud architecture

- **Cloud**: a large network of servers or indvidual PCs interconnetcted in a grid. They run in parallel and can combine their resources to generate a supercomputing-like power.
- **Intelligent management** to automatically connect computers, assign processing tasks, handle failures, etc.

Cloud storage

- data is stored on multiple servers $\Rightarrow$ users see a virtual server.
- Advantages: financially (cheaper thandedicated physical resources) and security (duplication managed by thrid party)

Cloud services

- Any web-based application or service offered via cloud computing is called a cloud service.
- The browser accesses the cloud service and an instance of the application is opened within the browser window.

Characteristics

- On-demand self service
- Resource pooling
- Rapid elasticity
- Measured service
- Broad network access

Service models

- IaaS
- PaaS
- SaaS

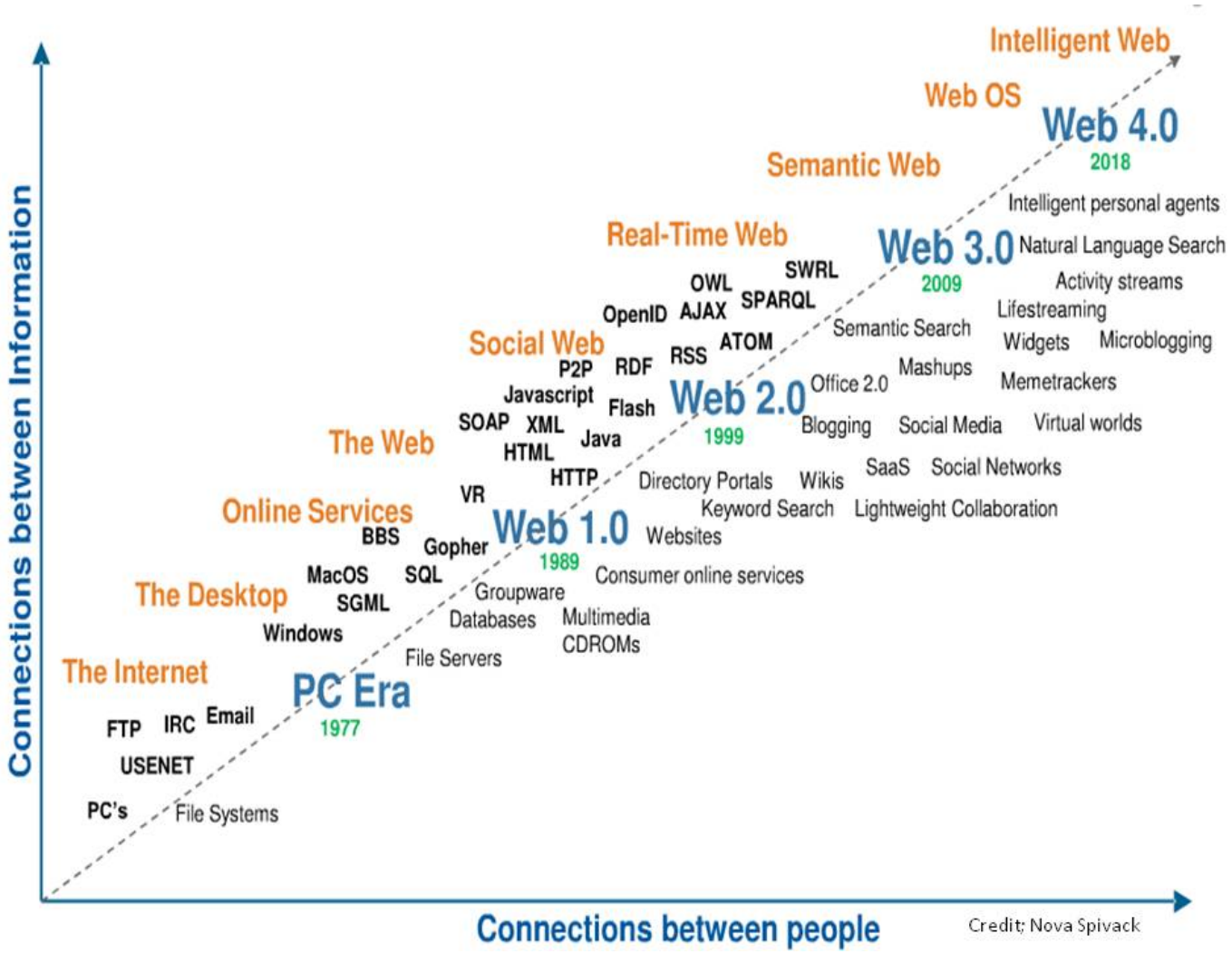
Deployment

- Public
- Private
- Community
- Hybrid (of public and private)

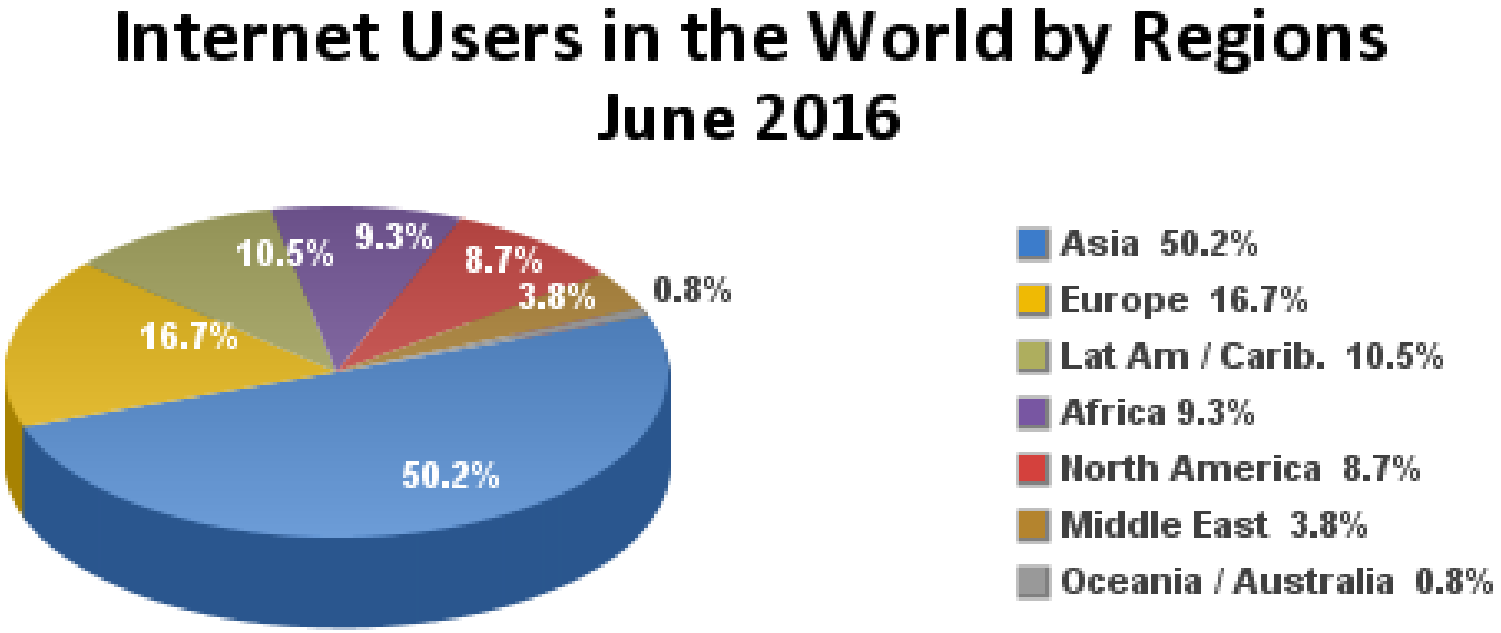
- Lower-cost of computers, infrastructure and softwares
- High performances
- Unlimited storage capacity
- Fewer maintenance
- Instant software updates
- Increased data safety
- Easier group collaboration
- Universal access to documents
- **Elasticity**
- **Automatization** of configuration tasks

- You need a constant internet connection
- Requires a large bandwidth
- Confidentiality of stored database
- May be more expensive for certain use case

Web evolution



Some statistics (credit to www.internetworldstats.com)



Source: Internet World Stats - www.internetworldstats.com/stats.htm
Basis: 3,675,824,813 Internet users on June 30, 2016

Some statistics (credit to www.internetworldstats.com)

WORLD INTERNET USAGE AND POPULATION STATISTICS JUNE 30, 2016 - Update						
World Regions	Population (2016 Est.)	Population % of World	Internet Users 30 June 2016	Penetration Rate (% Pop.)	Growth 2000-2016	Table % Users
Asia	4,052,652,889	55.2 %	**1,846,212,654**	45.6 %	1,515.2%	50.2 %
Europe	832,073,224	11.3 %	**614,979,903**	73.9 %	485.2%	16.7 %
Latin America / Caribbean	626,119,788	8.5 %	**384,751,302**	61.5 %	2,029.4%	10.5 %
Africa	1,185,529,578	16.2 %	**340,783,342**	28.7 %	7,448.8%	9.3 %
North America	359,492,293	4.9 %	**320,067,193**	89.0 %	196.1%	8.7 %
Middle East	246,700,900	3.4 %	**141,489,765**	57.4 %	4,207.4%	3.8 %
Oceania / Australia	37,590,820	0.5 %	**27,540,654**	73.3 %	261.4%	0.8 %
WORLD TOTAL	**7,340,159,492**	**100.0 %**	**3,675,824,813**	**50.1 %**	**918.3%**	**100.0 %**

Orders of magnitude of data

- kiloByte (kB) = 10^3
- megaByte (MB) = 10^6
- gigaByte (GB) = 10^9
- teraByte (TB) = 10^{12}, e.g. 20TB library of congress, Washington, USA (text only)
- petaByte (PB) = 10^{15}, e.g. 3PB library of congress, Washington, USA (text and images), 15PB :one year of storage of LHC
- exaByte (EB) = 10^{18}, e.g. 100EB information workflow in human brain over a lifetime (Von Neumann)
- zettaByte (SB) = 10^{21}
- yottaByte (YB) = 10^{24}

Application Service Provider (ASP) is a business that provides computer-based services to customers over a network.

- More satisfactory in terms of user friendliness
- Usually written in Java hence deploying a JRE is required
- Firewall problem due to client/server middleware

- In general, ASP applications are similar to classical enterprise applications, that is:
  - A unique application
  - With a single version available
  - A unique database
  - A single authentication system
- But ASP have N different clients. What if a client wants to tune its application? Does not want the last version of an app / database?
- Security issues: storing client data in the same database server!

- Rich client is a new opportunity for ASPs
- First appearing in 2003
- Is a mix between the client/server and Web environments.
- Decomposed in:
  - RIA: Rich Internet Application : Rich client based on a web browser
  - RDA: Rich Desktop Application : Rich client installed on the desktop. Deployed and updated via HTTP (see Windows update)

- No more deployment problems
- Limitations: handling a deconnected working mode. Everything is lost if we lose the network
- Some solutions:
  - Applet (Java)
  - Macromedia Flash/AIR
  - DHTML
  - Ajax
  - Silverlight (plateforme Microsoft)
  - JavaFX (plateforme Java)

- Pros:
  - Using Java APIs
  - Using data streaming, advanced GUIs and threading
- Cons:
  - Long loading times
  - A badly designed app can crash the web browser

- 1996
- ActionScript is the programming language
- Implementation examples: Flex, Lazlo (open source)
- Pros:
  - Can display vectorized images and animations
- Cons:
  - Web browser needs a plug-in
  - ActionScript is proprietary

- Dynamic HTML
- DHTML = Javascript + DOM + CSS
- Used for the creation of interactive apps
- Communication is asynchronous
- A complete page reload is needed

- AJAX: Asynchronous Javascript and XML
- Examples:
  - Google Maps:
    - Map data are requested and downloaded in a non asynchronous way.
    - Other parts of the web page are not impacted, no loss of operational context.
  - Google Suggest (end of 2004): support typing the first letters of a word and to interactively provide some suggestions.
  - Gmail

- Created by Microsoft in 2007
- Pros: not limited to IE
- Cons: needs a plug-in

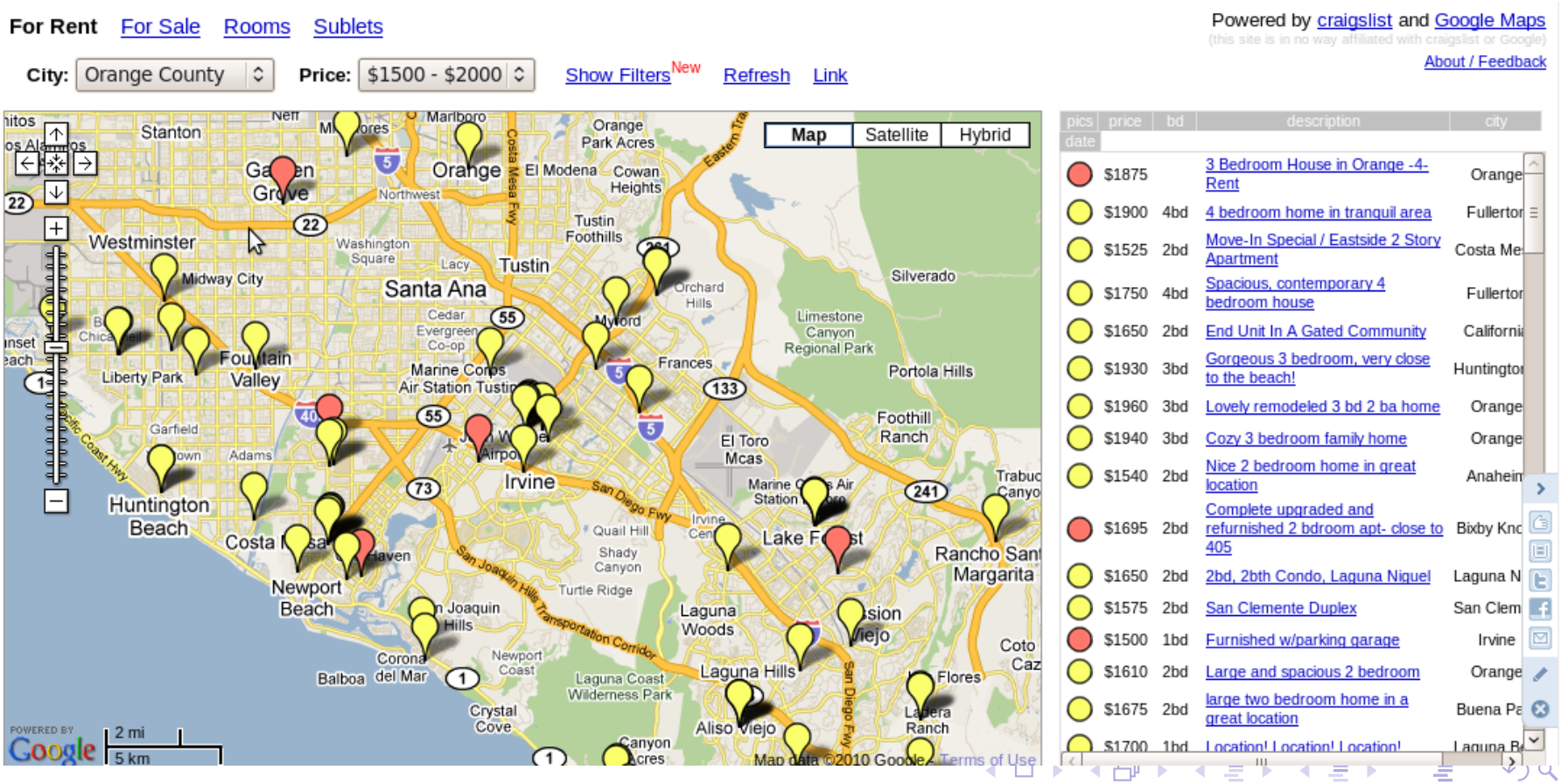
- RIA is the solution for ASP
- But their deconnected working mode is still an issue. Solutions:
  - Always on with 3G+ and wifi
  - Use Web browser extensions like embedded databases
  - New web browser natively handling a deconnected working mode
  - Use a synchronization app : ex: LiveMesh from Microsoft to synchronize your data with a server or another device.

- Buzz word appearing in 2005.
- Hard to define since it is a mix of new tools and emerging end-users behaviors.
- Web 1.0 involves a lot of read requests
- Web 2.0 involves even more reads but also a lot of writes

- Collective intelligence :
  - Wikipedia,
  - Book criticisms on Amazon,
  - CDDB music database
  - Blogs
  - Social networks
  - Picture and Video sharing (Flickr, YouTube)
  - Office apps on the Web (Google Spreadsheet)
  - End-users are more and more dependent on these apps
- End-users are more and more dependent on these apps

- An application combining contents or services of different apps
- In a Web context, a mashup aggregates contents coming several sites in order to create a new Web site with new functionalities
- Hence it requires an access to some data
- It is a design approach than a framework/took box
- In general uses Javascript and AJAX

www.housingmaps.com: Javascript mashup of Google maps and real estate from Craigslists



Design for mashup

- Separate content from presentation (MVC)
- Create APIs
- Let others knwo about your site and APIs
- Wait and study how people are using your APIs
- Evolve

Mashup issues

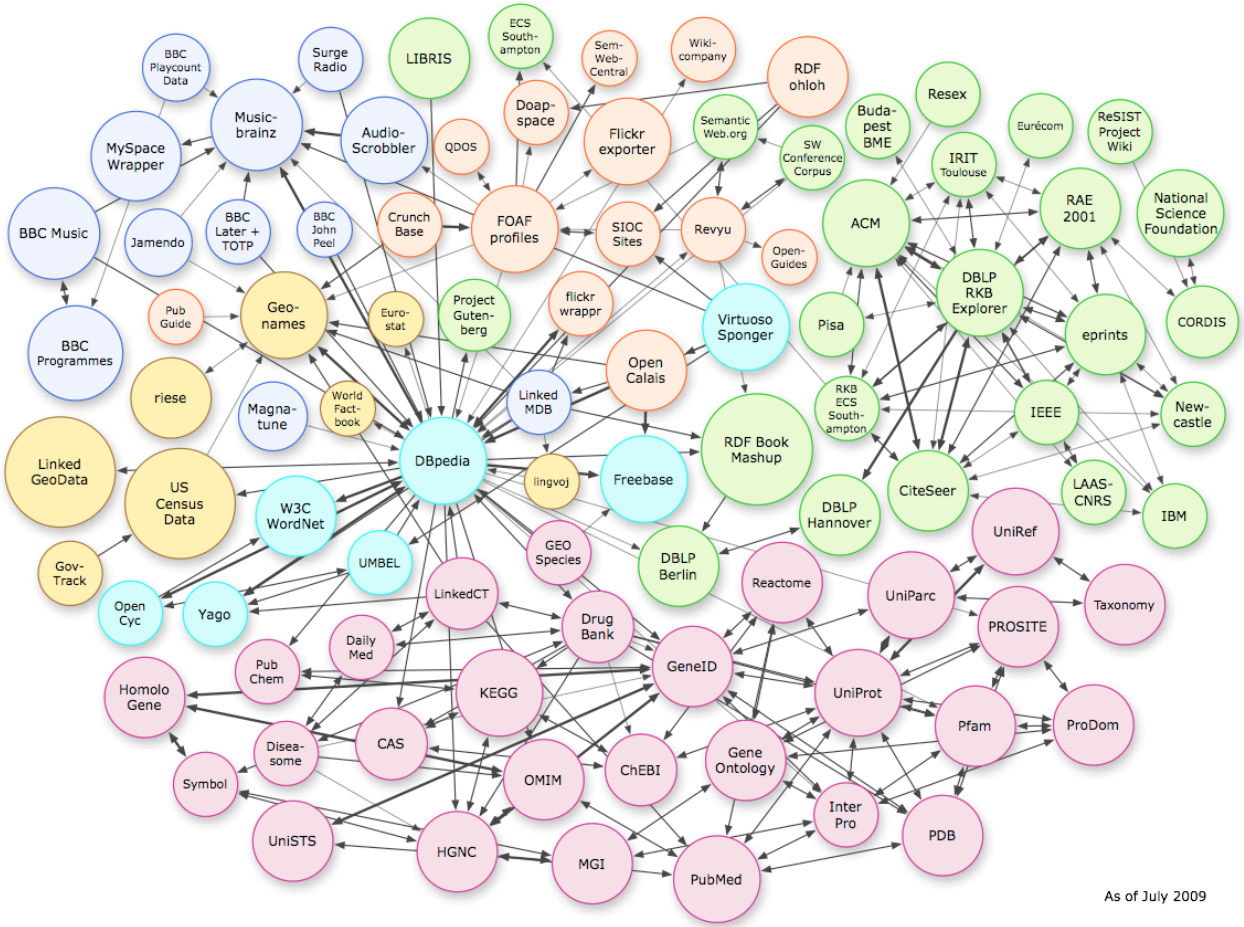
- What is integrated? Data format (JSON, HTML, RDF, etc.)
- Where is the integration being processed? Server vs client side
- How to get data? API vs Web scraping

Web scraping

- Write scripts to retrieve data from Web sites
- Examples: 339,000 places (incl. 241,000 populated places)

regional map	[show]
Time zone	CET (UTC +1)
Administration	
Country	France
Region	Île-de-France
Department	Paris (75)
Subdivisions	20 arrondissements
Mayor	Bertrand Delanoë (PS) (2008–2014)
Statistics	
Land area[1] [1]	1,118 km^2 (432 sq mi)
Population[2]	2,203,817 (January 1, 2009 estimate[2])
- Ranking	1st in France
Urban spread	
Urban area	2,723 km^2 (1,051 sq mi) (1999)
- Population	10,142,983[3] (2006)
Metro area	14,518.3 km^2 (5,605.5 sq mi) (1999)
- Population	11,769,433[4] (2006)
Website	paris.fr

Linked Data


BBC Playcount Data
Surge Radio
LIBRIS
ECS South-ampton
Sem-Web-Central
Wiki-company
RDF ohloh
Resex
Eurécom
ReSIST Project Wiki
Music-brainz
Audio-Scrobbler
Doap-space
Flickr exporter
Semantic Web.org
Buda-pest BME
IRIT Toulouse
MySpace Wrapper
QDOS
SW Conference Corpus
BBC Music
Jamendo
BBC Later + TOTP
BBC John Peel
Crunch Base
FOAF profiles
SIOC Sites
Revyu
Open-Guides
ACM
RAE 2001
National Science Foundation
Pub Guide
Geo-names
Euro-stat
Project Guten-berg
flickr wrappr
Virtuoso Sponger
Pisa
DBLP RKB Explorer
CORDIS
BBC Programmes
riese
Magna-tune
World Fact-book
Linked MDB
Open Calais
RKB ECS South-ampton
eprints
IEEE
New-castle
Linked GeoData
US Census Data
W3C WordNet
DBpedia
lingvoj
Freebase
RDF Book Mashup
CiteSeer
LAAS-CNRS
IBM
Gov-Track
UMBEL
GEO Species
DBLP Hannover
UniRef
Open Cyc
Yago
LinkedCT
DBLP Berlin
Reactome
UniParc
Taxonomy
Daily Med
Drug Bank
PROSITE
Pub Chem
GeneID
Homolo Gene
KEGG
UniProt
Pfam
ProDom
Disea-some
CAS
Gene Ontology
Symbol
OMIM
ChEBI
Inter Pro
UniSTS
HGNC
MGI
PubMed
PDB
As of July 2009

- Mobile devices reinforce Web 2.0
  - Tablets (iPad)
  - Ebooks (kindle)
  - Smartphones (iPhone, BlackBerry, Android)
- With their own limitations in terms of:
  - Ergonomy,
  - Technical environment (Linux, WindoIntroduction and motivation of cws,..)
  - Storage, CPU, Energy, Bandwidth

Some 2010 figures:

- Annual growth rate of 16.6 percent through 2015 for mobile internet.
- Mobile users to superpass desktop internet users in 2015
- Estimation of internet users in the world in 2015: 2.7 billion (40% of total world population)

- By providing a 'platform' for application hosting via the Web, cloud computing is the outcome of the different technologies presented
- But cloud computing goes further by providing more functionalities
- 2 figures:
  - How big is the world (of cloud computing)
  - Stats on cloud computing