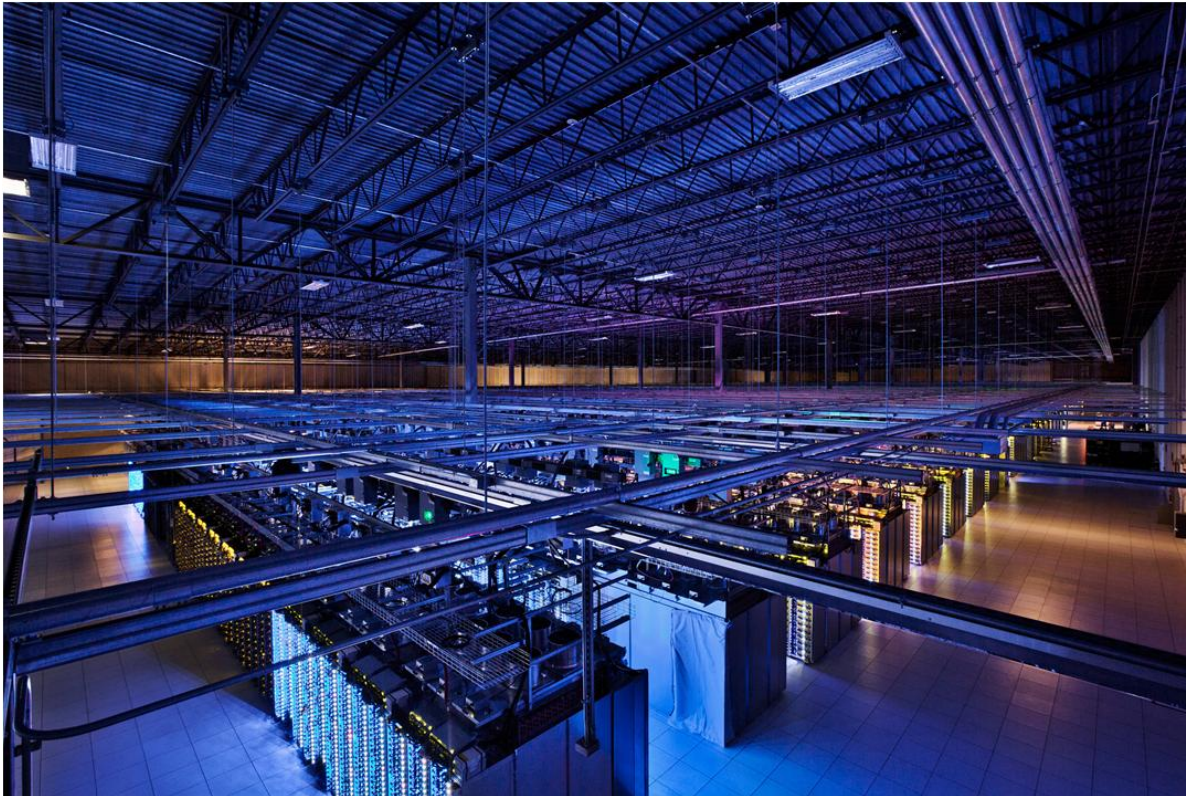


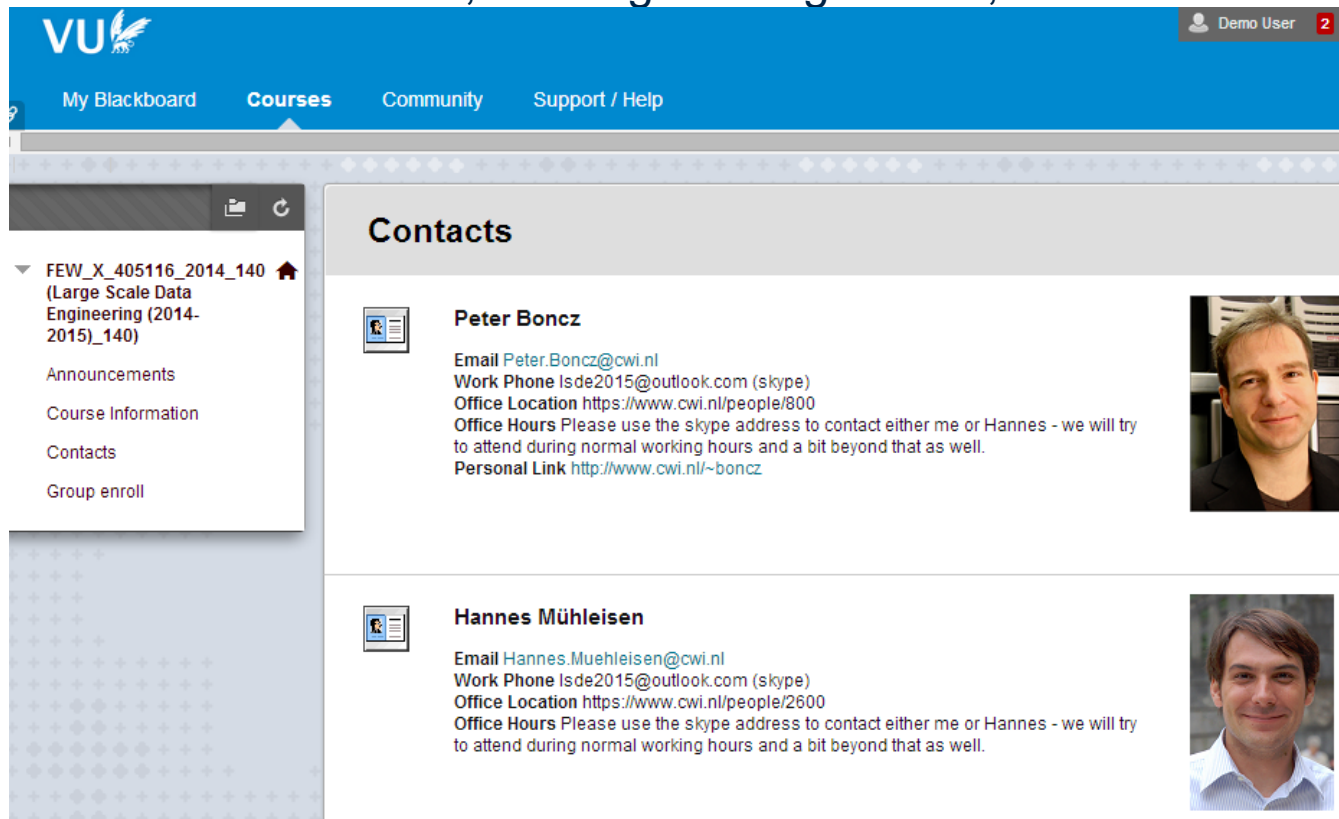
Large-Scale Data Engineering

Overview and Introduction



Administration

- Blackboard Page
 - Announcements, also via email (pardon html formatting)
 - Practical enrollment, Turning in assignments, Check Grades



The screenshot shows the Blackboard interface for a course. The top navigation bar is blue with the VU logo and a user profile for 'Demo User'. Below the navigation bar, the 'Contacts' page is displayed. On the left, a sidebar menu shows the course 'FEW_X_405116_2014_140 (Large Scale Data Engineering (2014-2015)_140)' with options for 'Announcements', 'Course Information', 'Contacts', and 'Group enroll'. The main content area is titled 'Contacts' and lists two contacts: Peter Boncz and Hannes Mühleisen. Each contact entry includes a profile picture, name, email, work phone, office location, office hours, and a personal link.

VU Demo User 2

My Blackboard Courses Community Support / Help

FEW_X_405116_2014_140 (Large Scale Data Engineering (2014-2015)_140)

Announcements

Course Information

Contacts

Group enroll

Contacts

Peter Boncz

Email Peter.Boncz@cwi.nl
Work Phone Isde2015@outlook.com (skype)
Office Location <https://www.cwi.nl/people/800>
Office Hours Please use the skype address to contact either me or Hannes - we will try to attend during normal working hours and a bit beyond that as well.
Personal Link <http://www.cwi.nl/~boncz>

Hannes Mühleisen

Email Hannes.Muehleisen@cwi.nl
Work Phone Isde2015@outlook.com (skype)
Office Location <https://www.cwi.nl/people/2600>
Office Hours Please use the skype address to contact either me or Hannes - we will try to attend during normal working hours and a bit beyond that as well.

- Contact: Email & Skype: Isde2015@outlook.com

Goals & Scope

- The goal of the course is to gain insight into and experience with algorithms and infrastructures for managing big data.
- Confronts you with some data management tasks, where
 - naïve solutions break down
 - problem size/complexity requires using a cluster
- Solving such tasks requires
 - insight in the main factors that underlie algorithm performance
 - access pattern, hardware latency/bandwidth
 - possess certain skills and experience in managing large-scale computing infrastructure.
 - slides and papers cover main cluster software infrastructures

What not to expect

- This course will NOT
 - Deal with High Performance Computing (exotic hardware etc.)
 - We deal with Cloud Computing, using commodity boxes
 - Deal with mobiles and how they can be cloud-enabled
 - They are simply the clients of the cloud just as any other machine is
 - Directly use commercial services
 - We try to teach industry-wide principles; vendor lock-in is not our purpose
 - Teach you how to program

Your Tasks

- Interact in class (always)
- Start working on Assignment 1 (now)
 - Form couples via Blackboard
 - Implement a ‘query’ program that solves a marketing query over a social network (and optionally also a ‘reorg’ program to store the data in a more efficient form).
 - Deadline within 2.5 weeks. Submit a *short* PDF report that explains what you implemented, experiments performed, and your final thoughts.
- Read the papers in the reading list as the topics are covered (from next week on)
- Pick a unique project for Assignment 2 (in 2.5 weeks)
 - 20min in-class presentation of your papers (last two weeks of lectures)
 - We can give presentation feedback beforehand (submit slides 24h earlier)
 - Conduct the project on a Hadoop Cluster (DAS-4 or SurfSARA)
 - write code, perform experiments
 - Submit a Project Report (deadline wk 13)
 - Related work (papers summary), Main Questions, Project Description, Project Results, Conclusion

Grading

- 30% Assignment1 (group grade)
- 20% Presentation (individual)
- 40% Assignment2 (group grade)
- 10% Attendance & interaction (individual)

What's on the menu?

1. Big Data
 - Why all the fuss?
2. Cloud computing infrastructure and introduction to MapReduce
 - What are the problems?
3. Hadoop MapReduce
 - Come play with the cool kids
4. Algorithms for Map Reduce
 - Oh, I didn't do much today, just programmed 10,000 machines
5. Replication and fault tolerance
 - Too many options are not always a good idea
6. NoSQL
 - The new “no” is the same as the old “no” but different
7. BASE vs. ACID
 - ...and other four-letter words
8. Data warehousing
 - Torture the data and it will confess to anything
9. Data streams
 - Being too fast too soon
10. Beyond MapReduce
 - Are we done yet? (No)

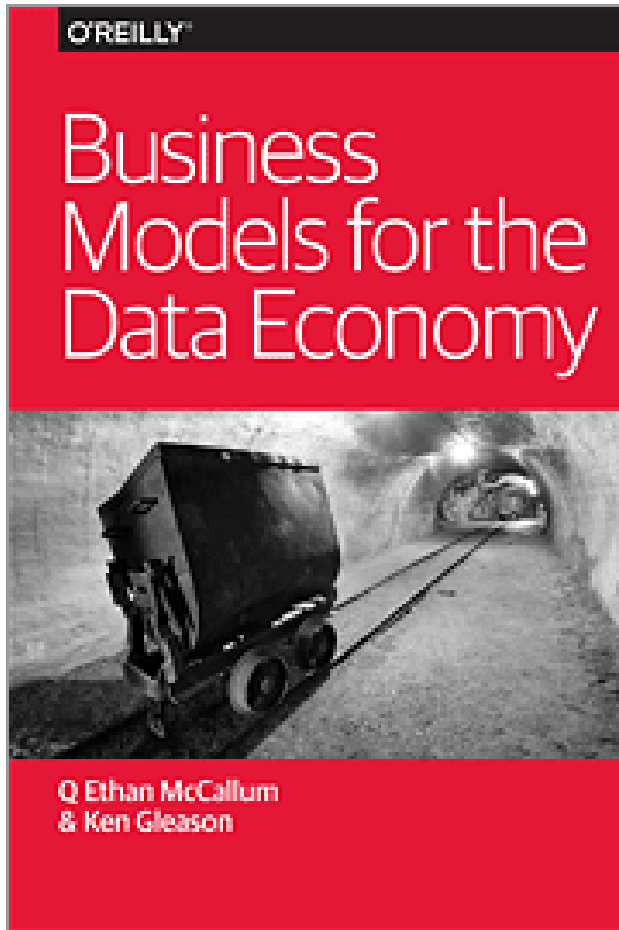
The age of Big Data



“Big Data”



The Data Economy

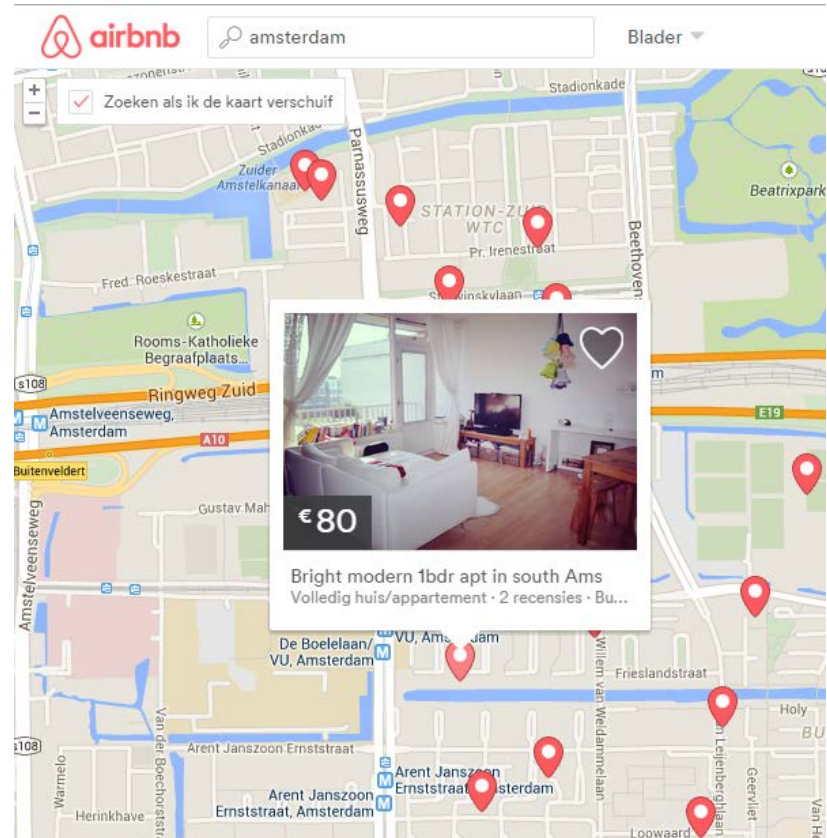
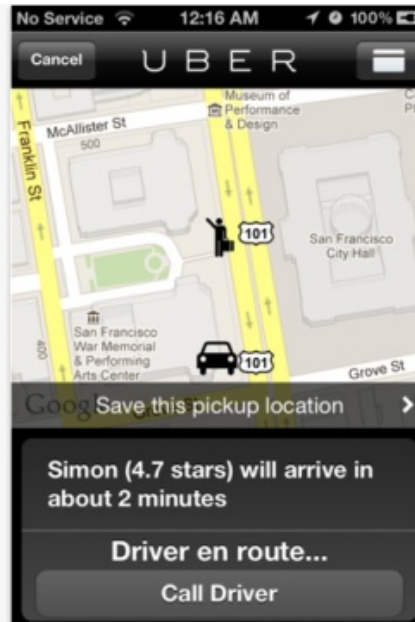
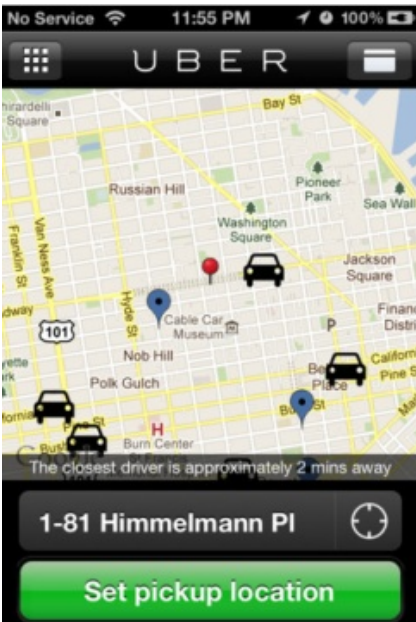


Disruptions by the Data Economy



U B E R

EVERYONE'S PRIVATE DRIVER™

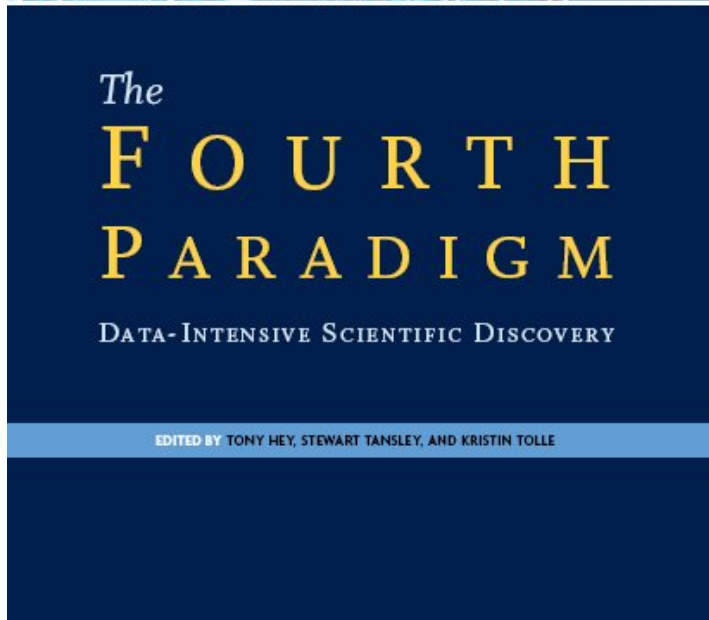


Data Disrupting Science



Scientific paradigms:

1. Observing
2. Modeling
3. Simulating
4. **Collecting and Analyzing Data**



Data Driven Science

International LOFAR Telescope (ILT)



Chilbolton



Opening van de LOFAR telescoop door koningin Beatrix in juni 2010

Dutch stations



raw data rate
30GB/sec
per station
=
1 full disk drive
per second

Large Scale Data Engineering



Big Data

- Big Data is a relative term
 - If things are breaking, you have Big Data
 - Big Data is not always Petabytes in size
 - Big Data for Informatics is not the same as for Google
- Big Data is often hard to understand
 - A model explaining it might be as complicated as the data itself
 - This has implications for Science
- The game may be the same, but the rules are completely different
 - What used to work needs to be reinvented in a different context

Power laws



- Big Data typically obeys a power law
- Modelling the head is easy, but may not be representative of the full population
 - Dealing with the full population might imply Big Data (e.g., selling all books, not just block busters)
- Processing Big Data might reveal power-laws
 - Most items take a small amount of time to process
 - A few items take a lot of time to process
- Understanding the nature of data is key

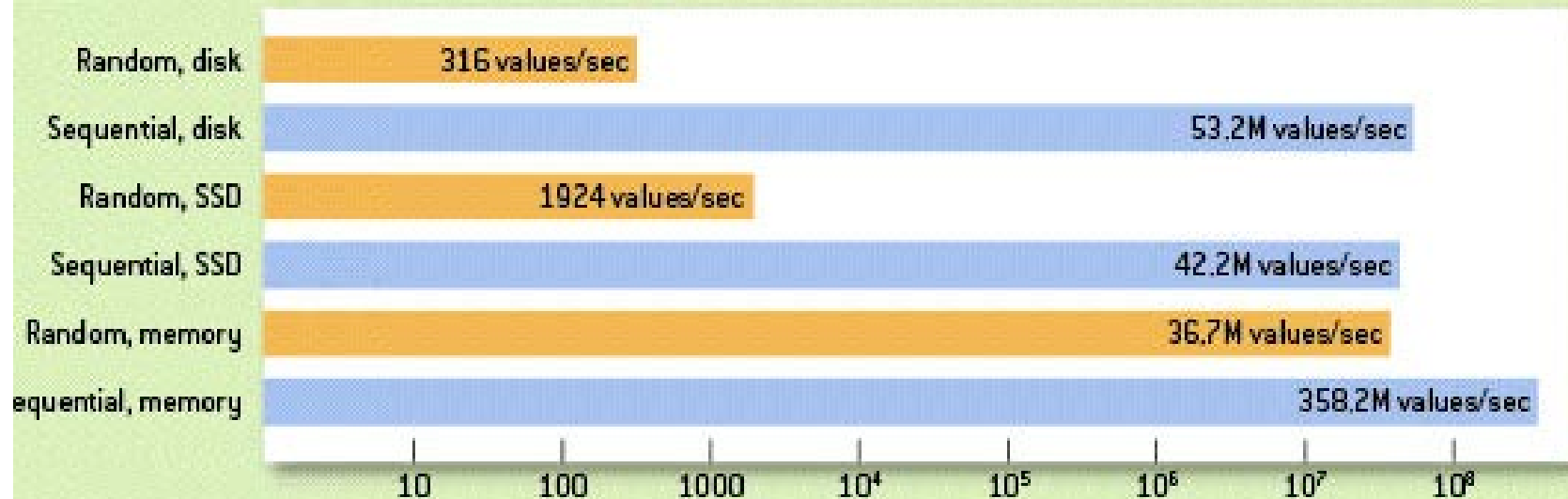
Big challenges: repeated observations

- Storing it is not really a problem: disk space is cheap
- Efficiently accessing it and deriving results can be hard
- Visualising it can be next to impossible
- Repeated observations
 - What makes Big Data big are repeated observations
 - Mobile phones report their locations every 15 seconds
 - People post on Twitter > 100 million posts a day
 - The Web changes every day
 - Potentially we need unbounded resources
 - Repeated observations motivates streaming algorithms

Big challenges: random access

3

Comparing Random and Sequential Access in Disk and Memory



Note: Disk tests were carried out on a freshly booted machine (a Windows 2003 server with 64-GB RAM and eight 15,000-RPM SAS disks in RAID5 configuration) to eliminate the effect of operating-system disk caching. SSD test used a latest-generation Intel high-performance SATA SSD.

Big challenges: denormalisation

- Arranging our data so we can use sequential access is great
- But not all decisions can be made locally
 - Finding the interest of my friend on Facebook is easy
 - But what if we want to do this for another person who shares the same friend?
 - Using random access, we would lookup that friend.
 - Using sequential access, we need to localise friend information
- Localising information means duplicating it
- Duplication implies denormalisation
- Denormalising data can greatly increase the size of it
 - And we're back at the beginning

Big challenges: non-uniform allocation

- Distributed computation is a natural way to tackle Big Data
 - MapReduce encourages sequential, disk-based, localised processing of data
 - MapReduce operates over a cluster of machines
- One consequence of power laws is uneven allocation of data to nodes
 - The head might go to one or two nodes
 - The tail would spread over all other nodes
 - All workers on the tail would finish quickly.
 - The head workers would be a lot slower
- Power laws can turn parallel algorithms into sequential algorithms

Big challenges: curation

- Big Data can be the basis of Science
 - Experiments can happen in silico
 - Discoveries can be made over large, aggregated data sets
- Data needs to be managed (curated)
 - How can we ensure that experiments are reproducible?
 - Whoever owns the data controls it
 - How can we guarantee that the data will survive?
 - What about access?
- Growing interest in Open Data

Economics and the pay-as-you-go model

- A major argument for Cloud Computing is pricing:
 - We could own our machines
 - ... and pay for electricity, cooling, operators
 - ...and allocate enough capacity to deal with peak demand
 - Since machines rarely operate at more than 30% capacity, we are paying for wasted resources
- Pay-as-you-go rental model
 - Rent machine instances by the hour
 - Pay for storage by space/month
 - Pay for bandwidth by space/hour
- No other costs
- This makes computing a commodity
 - Just like other commodity services (sewage, electricity etc.)

Bringing out the big guns

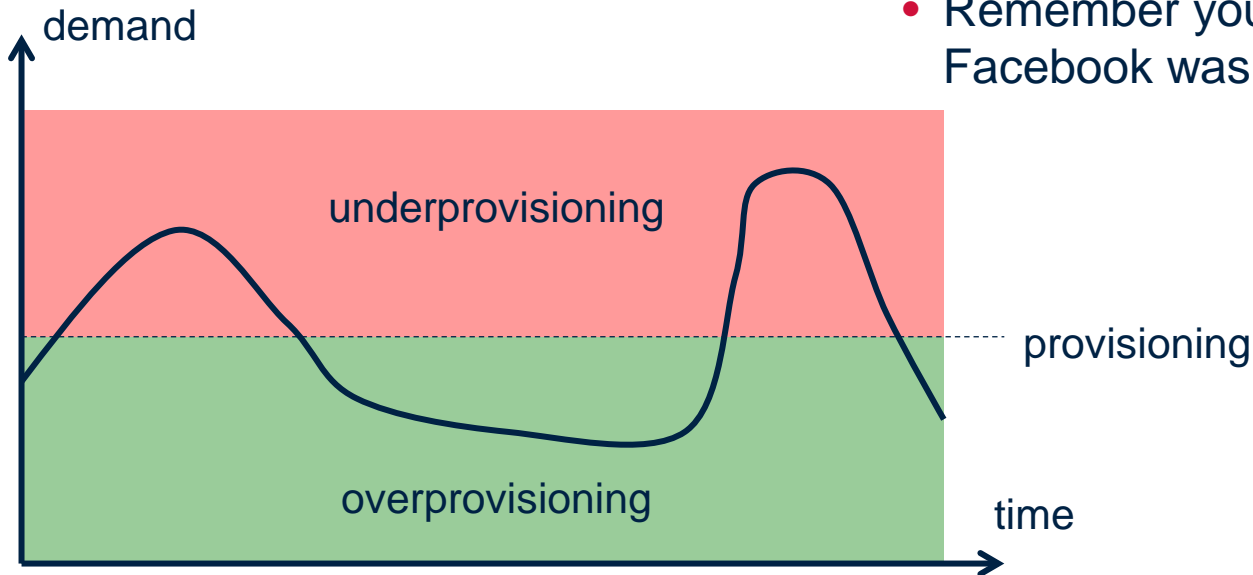
- Take the top two supercomputers in the world today
 - Tiahne-2 (Guangzhou, China)
 - Cost: US\$390 million
 - Titan (Oak Ridge National Laboratory, US)
 - Cost: US\$97 million
- Assume an expected lifetime of ten years and compute cost per hour
 - Tiahne-2: US\$4,110
 - Titan: US\$1,107
- This is just for the machine showing up at the door
 - Not factored in operational costs (e.g., running, maintenance, power, etc.)

Let's rent a supercomputer for an hour!

- Amazon Web Services charge US\$1.60 per hour for a large instance
 - An 880 large instance cluster would cost US\$1,408
 - Data costs US\$0.15 per GB to upload
 - Assume we want to upload 1TB
 - This would cost US\$153
 - The resulting setup would be #146 in the world's top-500 machines
 - Total cost: US\$1,561 per hour
 - Search for (first hit): LINPACK 880 server

Provisioning

- We can quickly scale resources as demand dictates
 - High demand: more instances
 - Low demand: fewer instances
- Elastic provisioning is crucial
- Target (US retailer) uses Amazon Web Services (AWS) to host target.com
 - During massive spikes (November 28 2009 – "Black Friday") target.com is unavailable
- Remember your panic when Facebook was down?



Data lock-in and third-party control

- Some provider hosts our data
 - But we can only access it using proprietary (non-standard) APIs
 - Lock-in makes customers vulnerable to price increases and dependent upon the provider
- Providers may control our data in unexpected ways:
 - July 2009: Amazon remotely remove books from Kindles
 - Twitter prevents exporting tweets more than 3200 posts back
 - Facebook locks user-data in
 - Paying customers forced off Picasa towards Google Plus
- Anti-terror laws mean that providers have to grant access to governments
 - This privilege can be overused

High performance and low latency

- How quickly data moves around the network
 - Total system latency is a function of memory, CPU, disk and network
 - The CPU speed is often only a minor aspect
- Examples
 - Algorithmic Trading (put the data centre near the exchange); whoever can execute a trade the fastest wins
 - Simulations of physical systems
 - Search results
 - Google 2006: increasing page load time by 0.5 seconds produces a 20% drop in traffic
 - Amazon 2007: for every 100ms increase in load time, sales decrease by 1%
 - Google's web search rewards pages that load quickly

Privacy and security

- People will not use Cloud Computing if trust is eroded
 - Who can access it?
 - Governments? Other people?
 - Snowden is the Chernobyl of Big Data
 - Privacy guarantees needs to be clearly stated and kept-to
- Privacy breaches
 - Numerous examples of Web mail accounts hacked
 - Many many cases of (UK) governmental data loss
 - TJX Companies Inc. (2007): 45 million credit and debit card numbers stolen
 - Every day there seems to be another instance of private data being leaked to the public

Summary

- Introduced the notion of Big Data
- Looked at various challenges
- Motivated some of the later techniques
- Computing as a commodity is likely to increase over time
- Cloud Computing adaptation and adoption are driven by economics
- The risks and obstacles behind it are complex