



ZooKeeper Tutorial

Flavio Junqueira
Benjamin Reed

Yahoo! Research

<https://cwiki.apache.org/confluence/display/ZOOKEEPER/Eurosystutorial>

Plan for today

- First half
 - Part 1
 - Motivation and background
 - Part 2
 - How ZooKeeper works on paper
- Second half
 - Part 3
 - Share some practical experience
 - Programming exercises
 - Part 4
 - Some caveats
 - Wrap up





ZooKeeper Tutorial

Part 1
Fundamentals

Yahoo! Portal

The diagram illustrates the Yahoo! Portal interface with several modules:

- Search**: Located at the top right, connected to the search bar.
- E-mail**: Located below the search bar, connected to the Mail icon in the Personal Assistant module.
- Finance**: Located to the right of the Mail preview, connected to the Stocks icon in the Personal Assistant module.
- Weather**: Located below Finance, connected to the Weather icon in the Personal Assistant module.
- News**: Located at the bottom right, connected to the News module in the Yahoo! News: Most Viewed section.

The main interface includes:

- Personal Assistant**: Contains icons for Mail, Horoscope, Stocks, Weather (57°F), Lottery, and Sports.
- Message Center**: Contains a Weather section for Philadelphia, PA, showing mostly cloudy conditions with temperatures of 70° / 49°, 61° / 42°, and 63° / 47° for Today, Tomorrow, and Monday respectively. It also shows a severe weather alert for Barcelona, Spain.
- Web Search**: A yellow button in the top navigation bar.
- Yahoo! Noticias: Foto de Portada**: Headline: "PESHAWAR, Pakistán (AFP) - Ataque suicida y primera victoria en ofensiva en el sur de Pakistán".
- Yahoo Mail Preview**: Preview for "fpjunqueira" showing travel information for a flight to Frankfurt/Main, travel news from Lufthansa, and deals from Travelocity and Economist.com.
- Yahoo News: Most Viewed**: Headline: "Church janitor charged in slaying of NJ priest (AP)".



Yahoo!: Workload generated

- Home page
 - 38 million users a day (USA)
 - 2.5 billion users a month (USA)
- Web search
 - 3 billion queries a month
- E-mail
 - 90 million actual users
 - 10 min/visit



Yahoo! Infrastructure

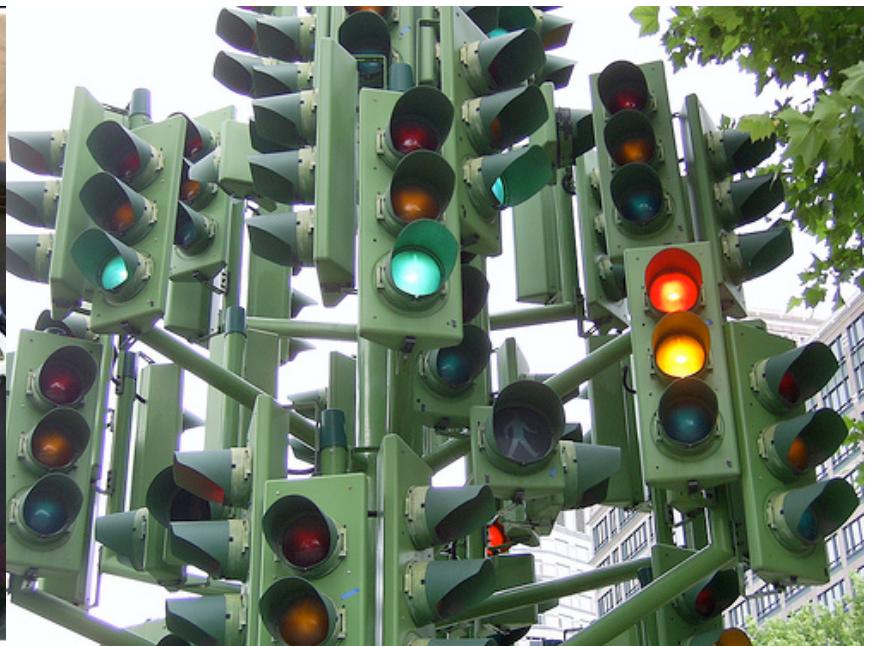
- Lots of servers
- Lots of processes
- High volumes of data
- Highly complex software systems
- ... and developers are mere mortals



Yahoo! Lockport Data Center



Coordination is important



Coordination primitives

- Semaphores
- Queues
- Leader election
- Group membership
- Barriers
- Configuration

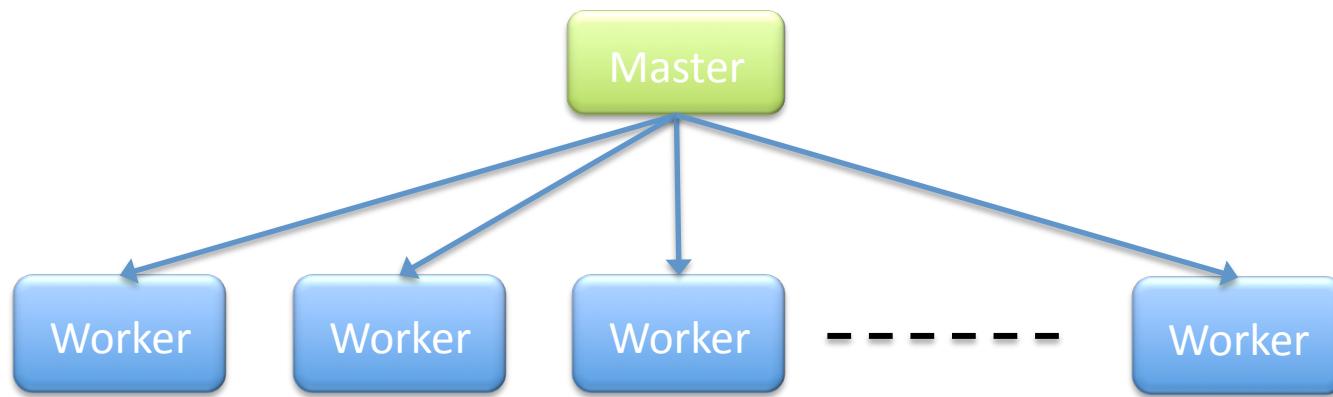


Even small is hard...



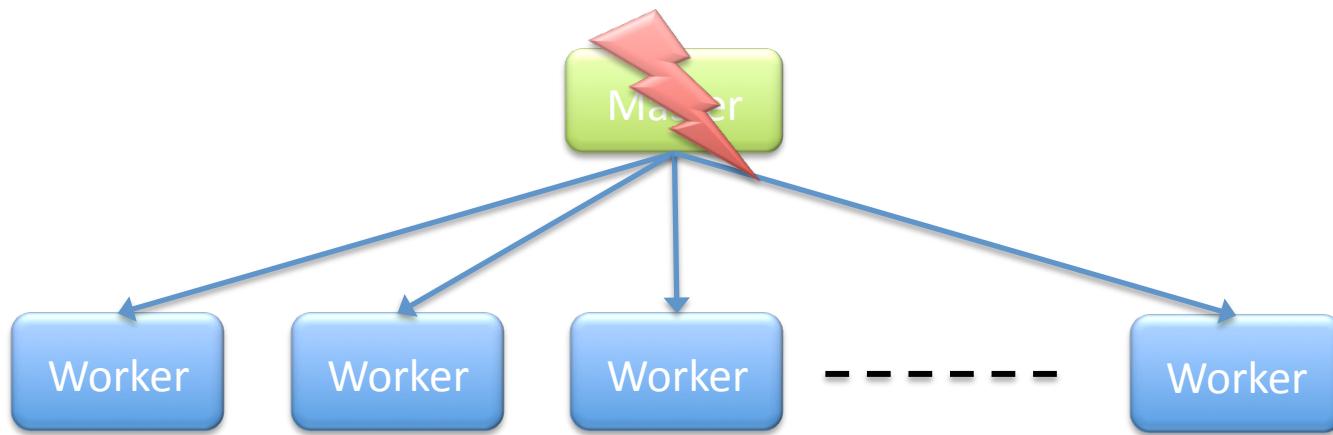
A simple model

- Work assignment
 - Master assigns work
 - Workers execute tasks assigned by master



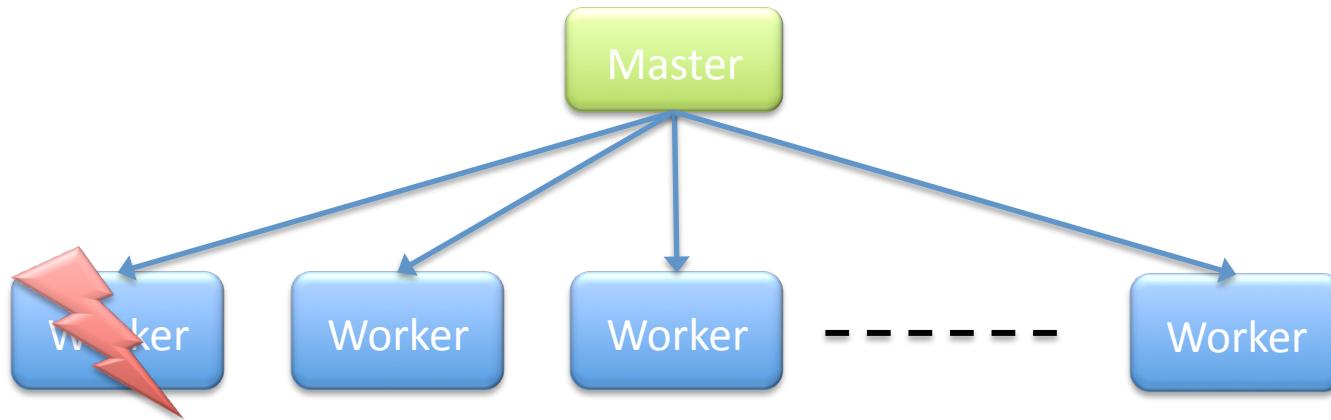
Master crashes

- Single point of failure
- No work is assigned
- Need to select a new master



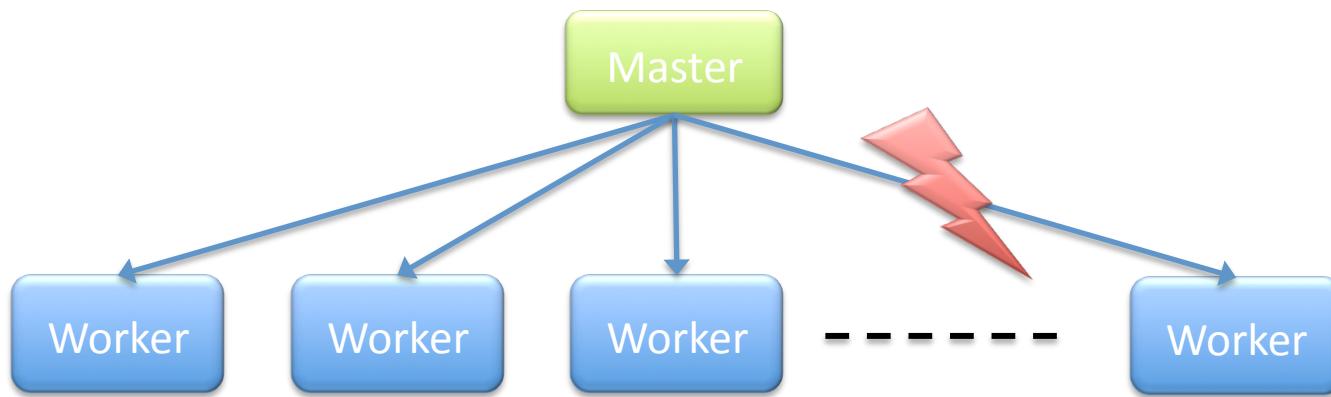
Worker crashes

- Not as bad... Overall system still works
 - Does not work if there are dependencies
- Some tasks will never be executed
- Need to detect crashed workers

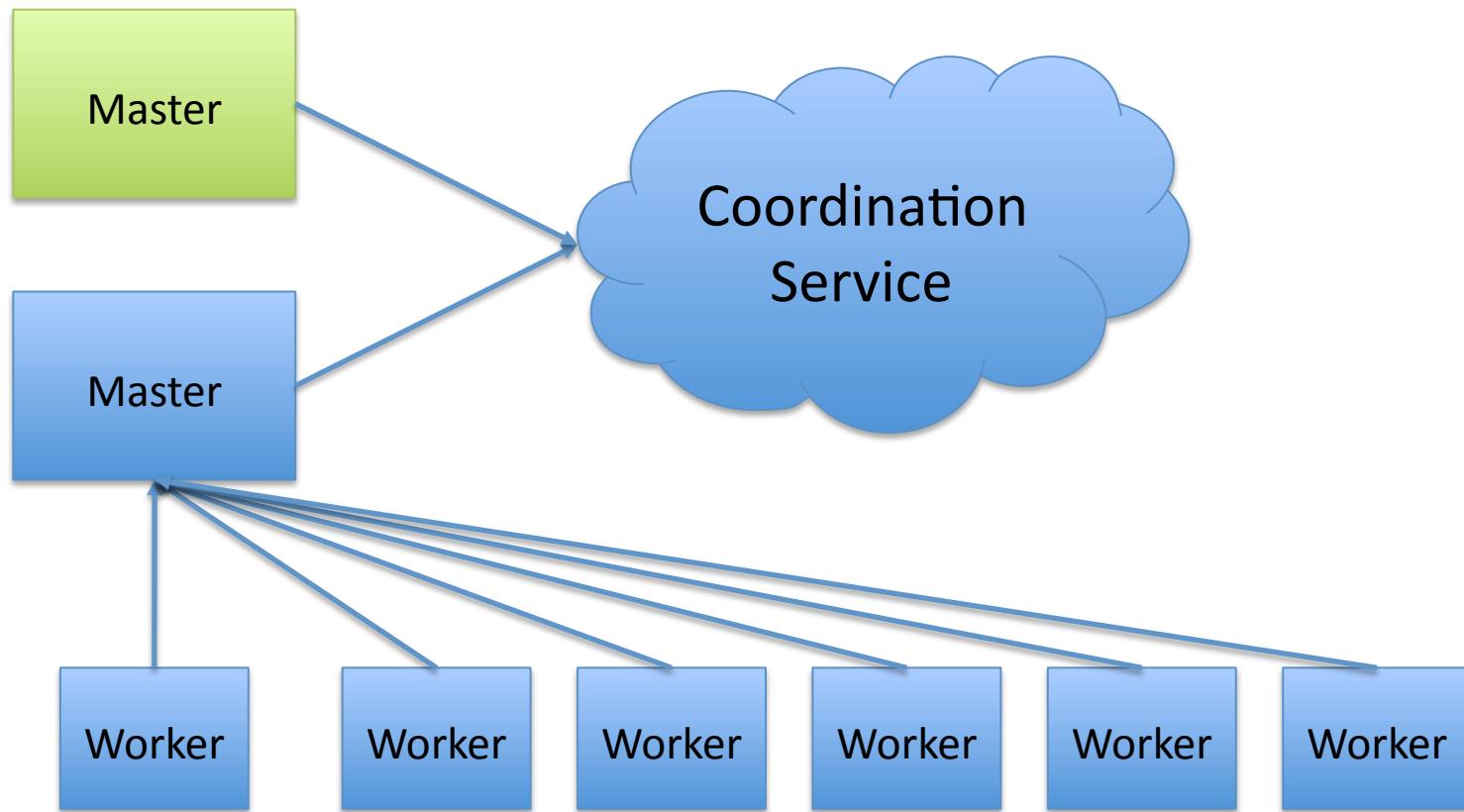


Worker does not receive assignment

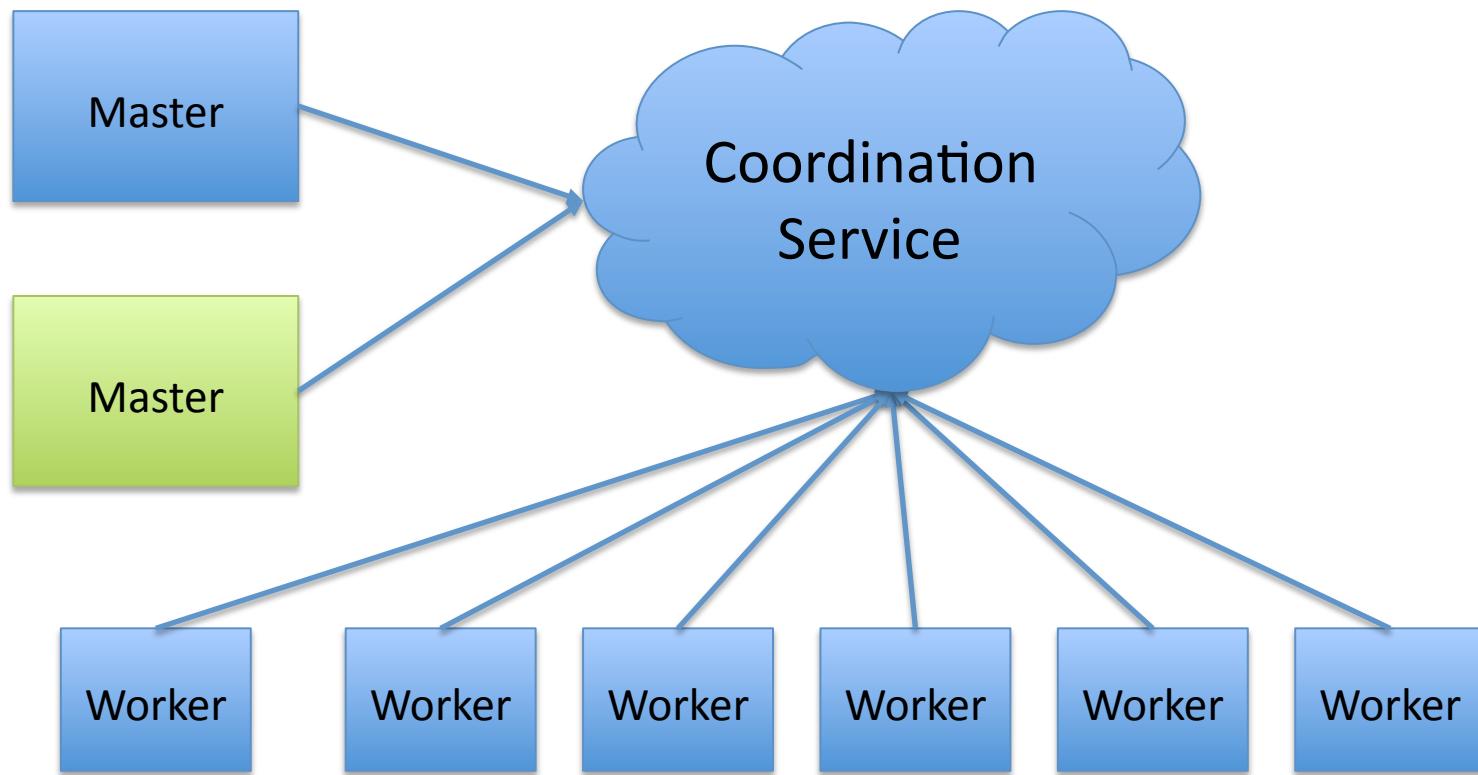
- Same problem as before
- Some tasks may not be executed
- Need to guarantee that worker receives assignment



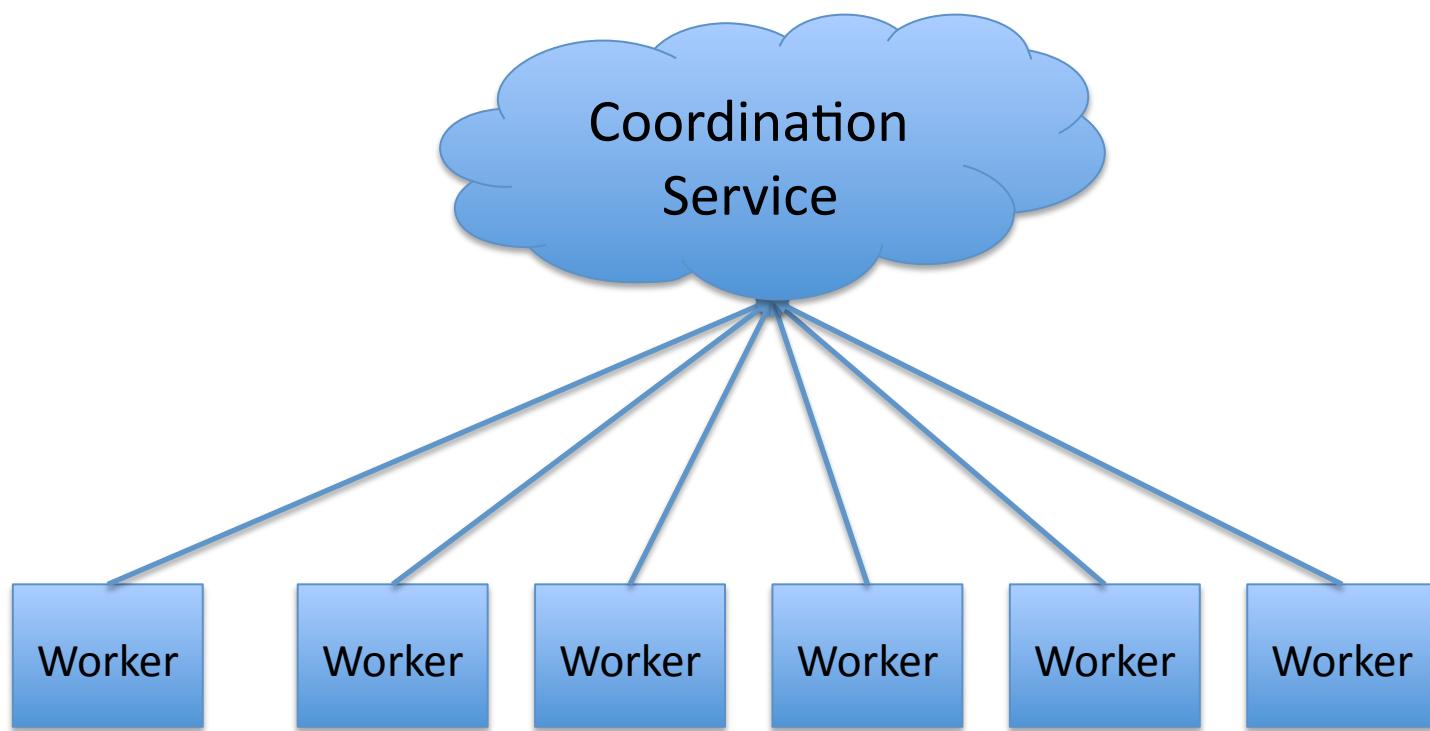
Fault-tolerant distributed system



Fault-tolerant distributed system



Fully distributed



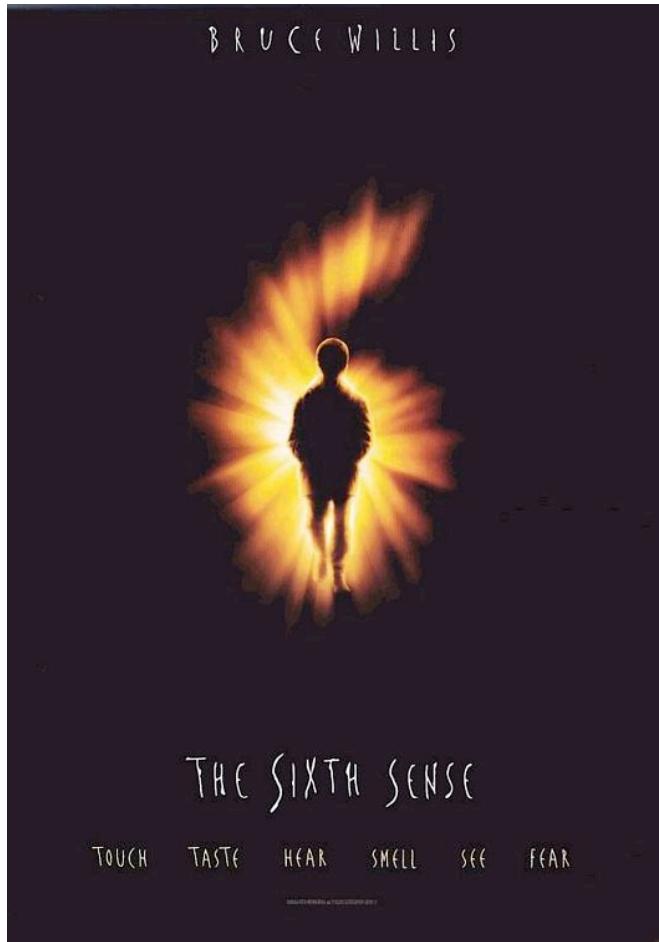
Fallacies of distributed computing

1. The network is reliable.
2. Latency is zero.
3. Bandwidth is infinite.
4. The network is secure.
5. Topology doesn't change.
6. There is one administrator.
7. Transport cost is zero.
8. The network is homogeneous.

Peter Deutsch, <http://blogs.sun.com/jag/resource/Fallacies.html>



One more fallacy



- You know who is alive



Why is it difficult?

- FLP impossibility result
 - Asynchronous systems
 - Consensus is impossible if a single process can crash

Fischer, Lynch, Paterson, ACM PODS, 1983

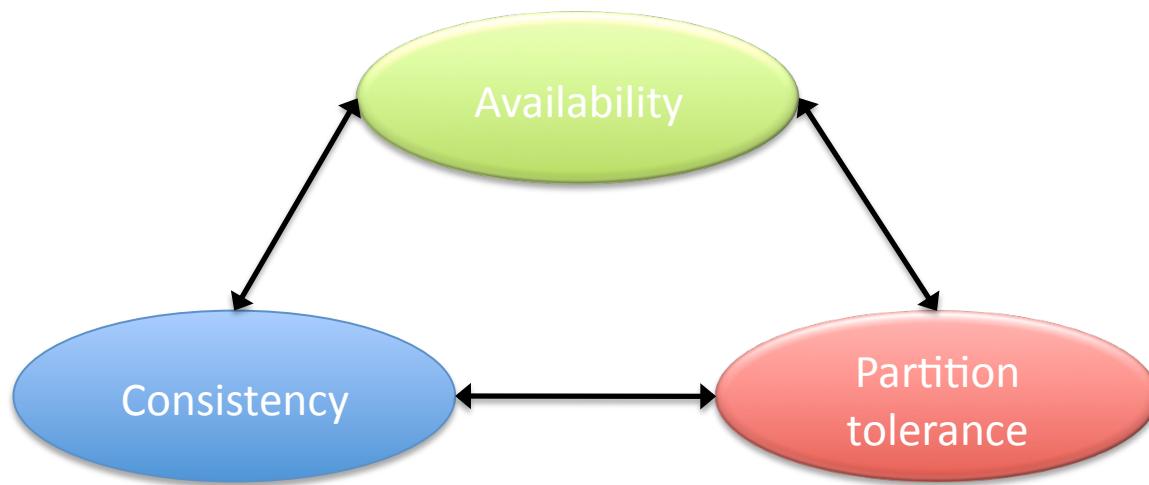
- According to Herlihy, we do need consensus
 - Wait-free synchronization
 - Wait-free: completion in a finite number of steps
 - Universal object: equivalent to solving consensus for n processes

Herlihy, ACM TOPLAS, 1991



Why is it difficult?

- CAP principle
 - Can't obtain availability, consistency, and partition tolerance simultaneously



Gilbert, Lynch, ACM SIGACT NEWS, 2002



The case for a coordination service

- Many impossibility results
- Many fallacies to stumble upon
- Several common requirements across applications
 - Duplicating is bad
 - Duplicating poorly is even worse
- Coordination service
 - Implement it once and well
 - Share by a number of applications



Current systems

- Chubby, Google
 - Lock service

Burrows, USENIX OSDI, 2006

- Centrifuge, Microsoft
 - Lease service

Adya et al., USENIX NSDI, 2010

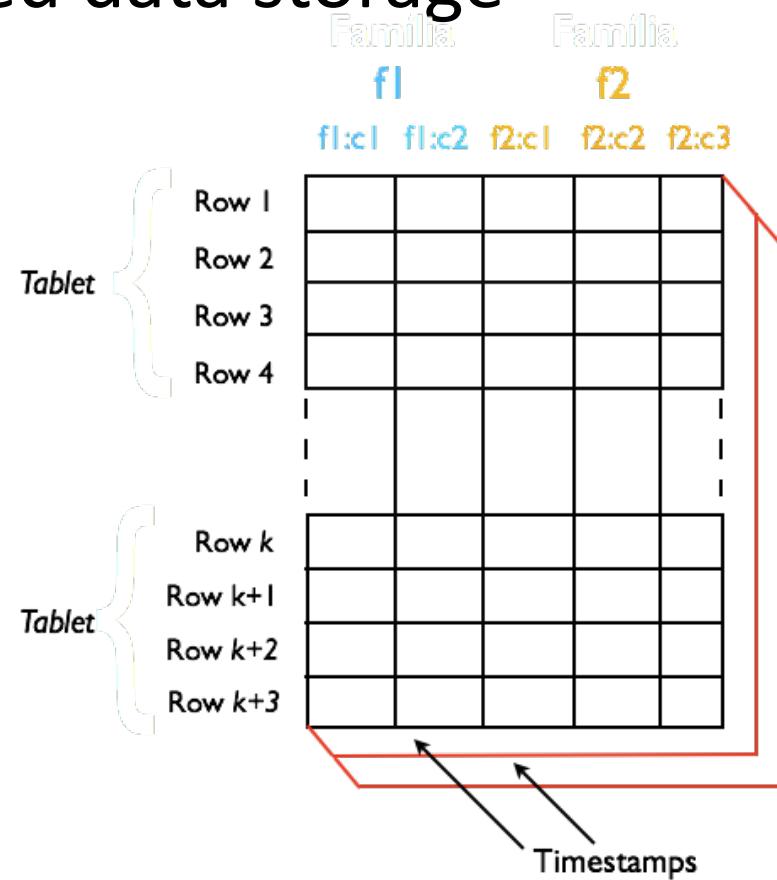
- ZooKeeper, Yahoo!
 - Coordination kernel
 - On Apache since 2008

Hunt et al., USENIX ATC, 2010



Example – Bigtable, HBase

- Sparse column-oriented data storage
 - Tablet: range of rows
 - Unit of distribution
- Architecture
 - Master
 - Tablet servers



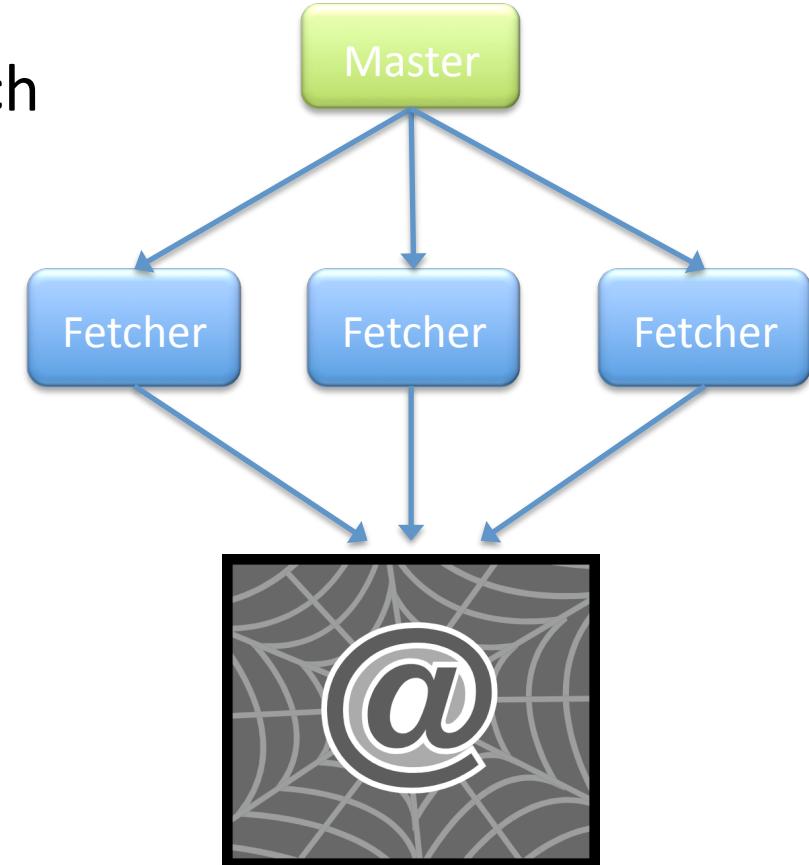
Example – Bigtable, HBase

- Master election
 - Tolerate master crashes
- Metadata management
 - ACLs, Tablet metadata
- Rendezvous
 - Find tablet server
- Crash detection
 - Live tablet servers



Example – Web crawling

- Fetching service
 - Fetch Web pages for search engine
- Master election
 - Assign work
- Metadata management
 - Politeness constraints
 - Shards
- Crash detection
 - Live workers



And more examples...

- GFS – Google File System
 - Master election
 - File system metadata
- Katta - Document indexing system
 - Shard information
 - Index version coordination
- Hedwig – Pub-Sub system
 - Topic metadata
 - Topic assignment



Summary of Part 1

- Large infrastructures require coordination
- Fallacies of distributed computing
- Theory results: FLP, CAP
- Coordination services
- Examples
 - Web search
 - Storage systems

