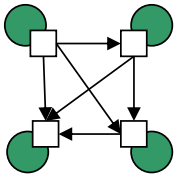
U LISBOA | UNIVERSIDADE DE LISBOA

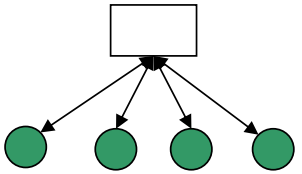
The Future of Dependable Distributed Systems is Simple

Alysson Bessani

(Example 1)
Group Communication vs. Coordination Services

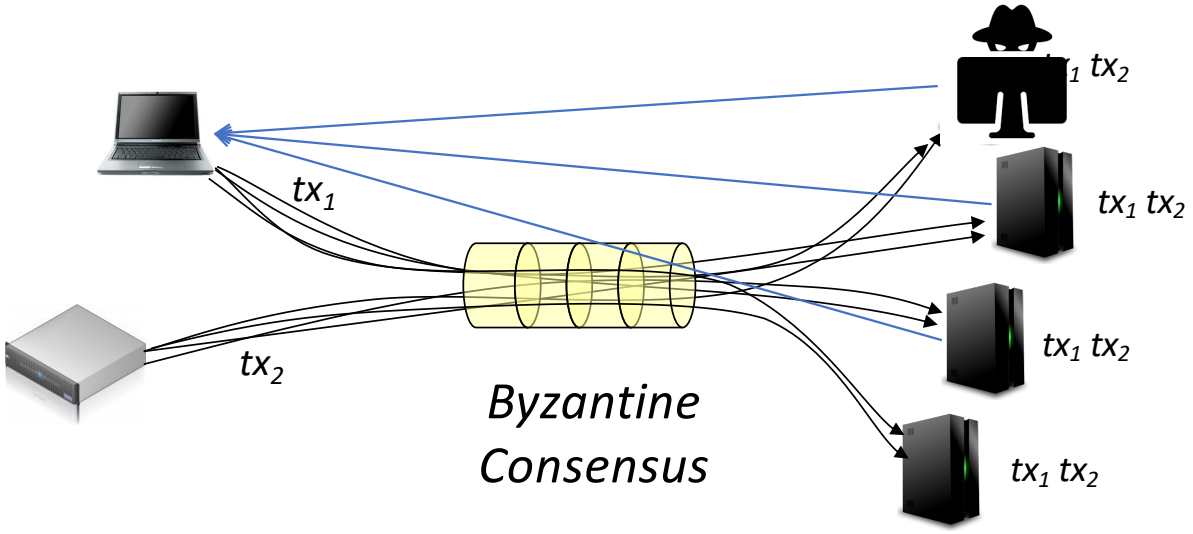


- Standard way to coordinate processes in the 90's
- Focus on QoS composability
- Deemed difficult to use by "normal" programmers
  - Unnatural programming model
  - Not very scalable
- Not the right abstraction for coordination



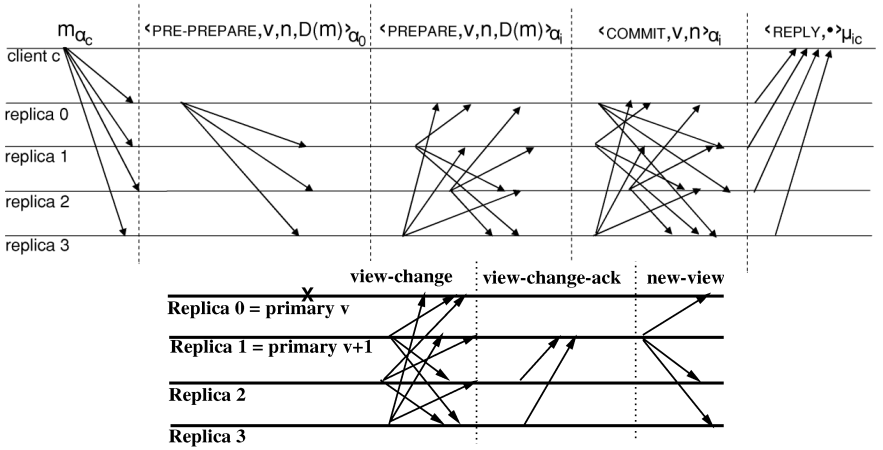
- Complex distributed algorithms encapsulated inside a "server"
- Server-based programming model resembling shared memory with synchronization power
- Single service shared by many apps and thousands of clients
- Widely used: ZooKeeper, Etcd, …

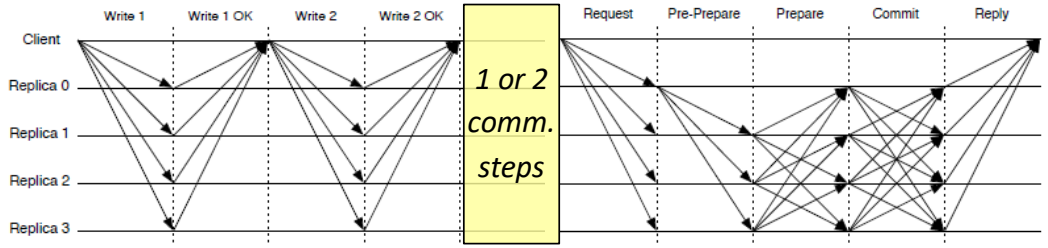
(Example 2)
Practical Byzantine Fault Tolerant (BFT) Systems



1st generation: Fast & Complex

- PBFT [OSDI'99] is considered the first "practical" BFT protocol
  - Fast due to the avoidance of public key signatures (expensive at the time)
- Several other works tried to improve the performance of PBFT by favorable expected common cases
  - HQ-Replication [OSDI'06] ->
- These protocols were fragile and complex



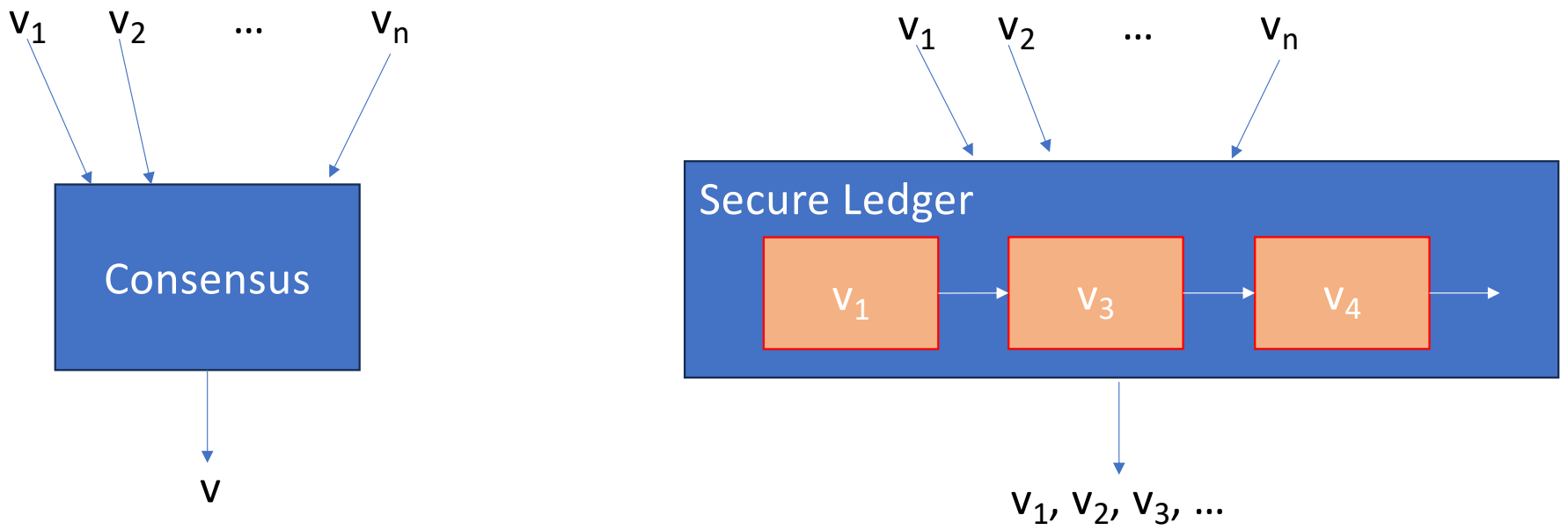


2nd generation: Resilient and Modular

- Use of trusted components
  - A2M [SOSP'07], MinBFT [TC'13], CheapBFT [EuroSys'12], Hybster [EuroSys'17]
- Resilient to simple and sophisticated attacks
  - Prime [DSN'08], Aardvark [NSDI'09], Spinning [SRDS'09], RBFT [ICDCS'13]
- Modular
  - Abstract [EuroSys'10, ToCS'15]
- Robust implementation
  - BFT-SMaRt [DSN'14]

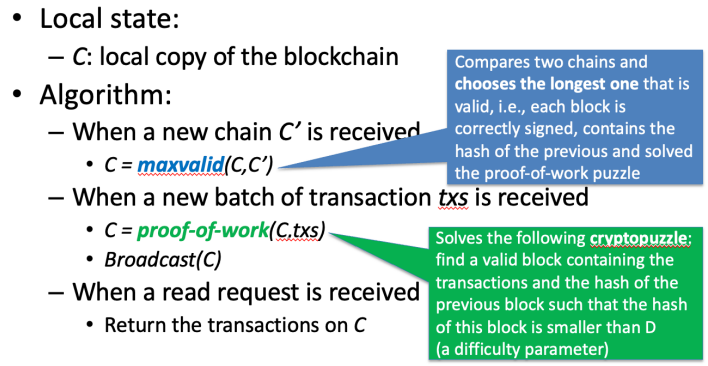
3rd generation: Blockchain inspired

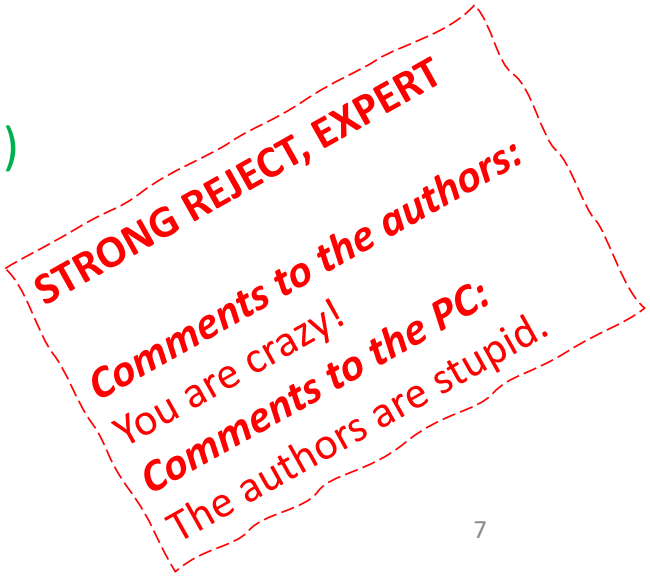
- Change of abstractions



3rd generation: Blockchain inspired

- Nakamoto consensus (used in Bitcoin and many other systems)
  - Slow (few transactions per second; latency of tens of minutes)
  - Uses all the CPU that is available in the system
  - Synchronous
  - Requires infinite storage
  - No clear resilience threshold
  - Scalable (performance is independent of the system size)
  - Simple

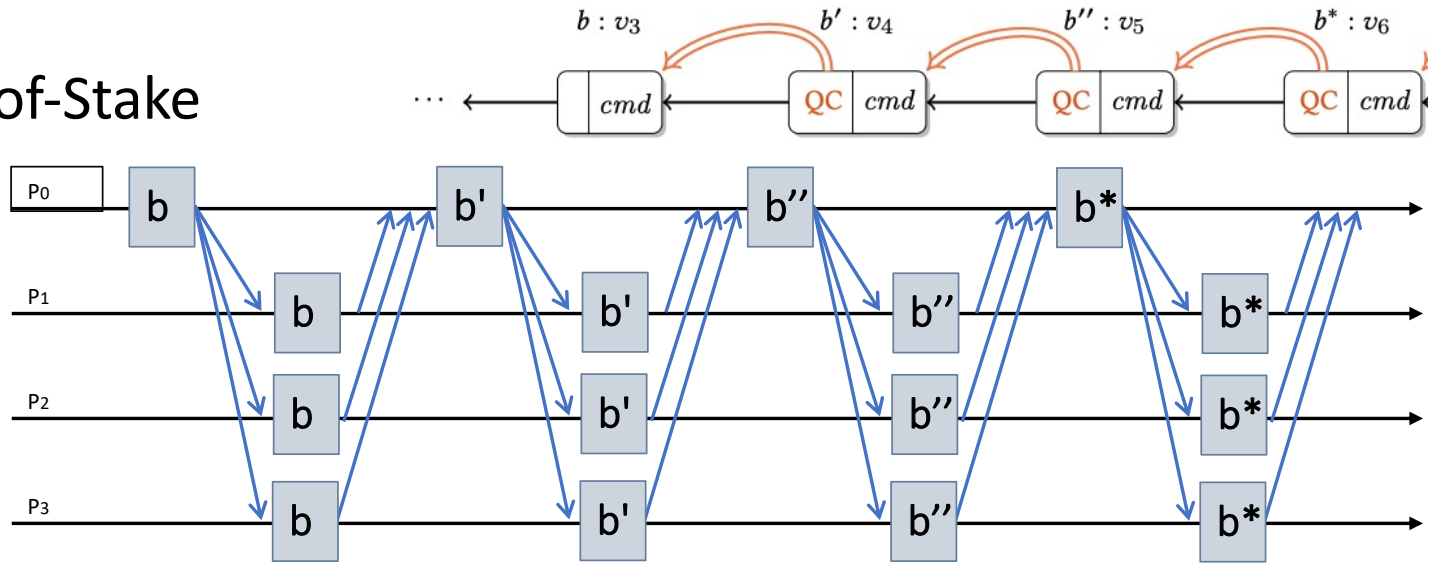




3rd generation: Blockchain inspired

- Chained consensus
  - Simple block approval protocol
  - Commit rule
  - Chain selection rule
- Used in most Proof-of-Stake blockchains

HotStuff [PODC'09]



The Simple Approach

- The cost of simplicity
  - Introduce assumptions
    - Synchronous communication
    - Synchronized clocks
    - Limited adversarial power
  - Sacrifice properties
    - Liveness and latency suffer
- But
  - It is understandable
  - Less prone to unknown weaknesses
  - Easier to formally verify

- The role of complexity
  - Important for reaching simplicity
  - Sometimes unavoidable
  - Enable huge benefits

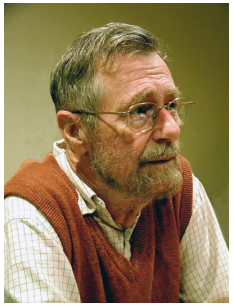
- How to do that?
  - New abstractions
  - Modularity
  - Transformations
  - Compositions

The Future DDS is simple!

```
>>> Distributed computing real world impact is
driven by simplicity.
```

or

```
>>> Simple designs always win.
```



Simplicity is prerequisite for reliability.

Edsger Dijkstra

Questions?

- Alysson Bessani
  - anbessani@fc.ul.pt
  - www.di.fc.ul.pt/~bessani

This work is partially supported by FCT through the ThreatAdapt project (FCT-FNR/0002/2018), and the LASIGE Research Unit (UIDB/00408/2020 and UIDP/00408/2020).