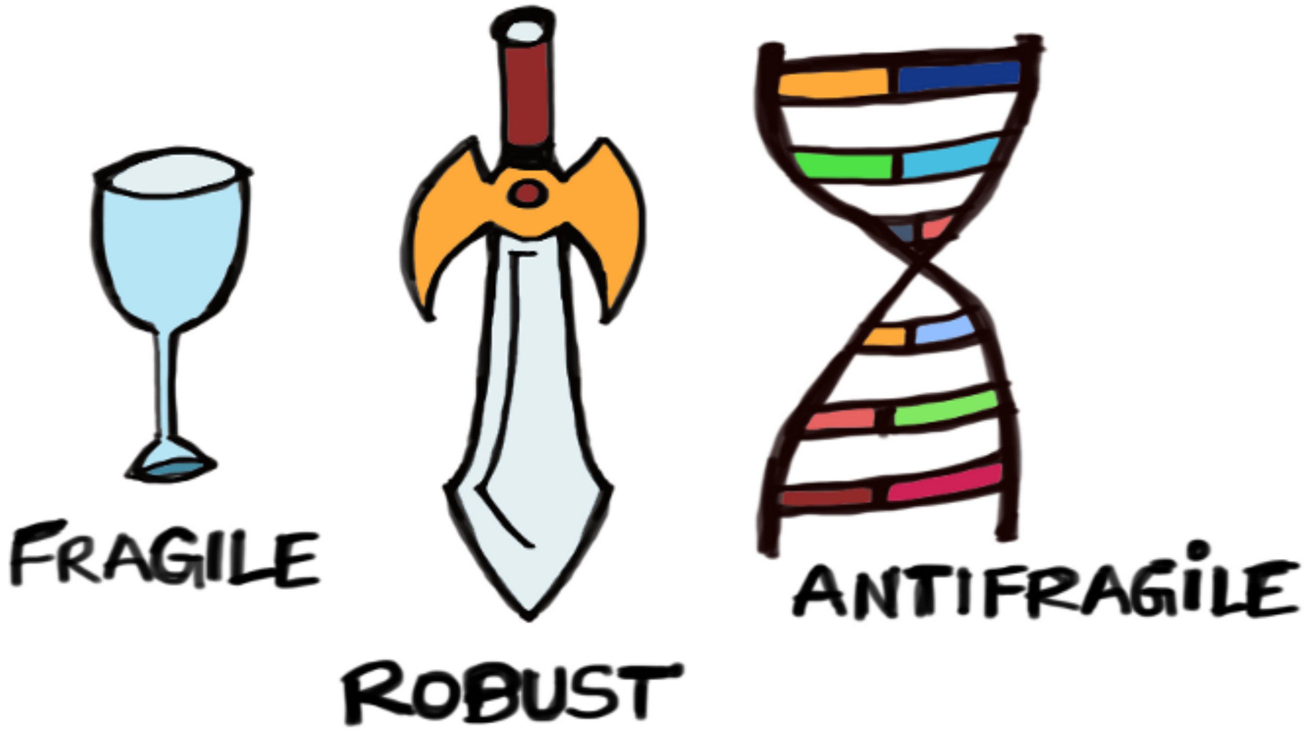
Challenging Anti-fragile Blockchain systems

Miguel González
Univ. Lille 1

What is Anti-fragile?



Harmed by disorder

Resilient to disorder

Benefits and Learns from disorder

Systems considered anti-fragile

- Financial system
- Human Body
- Restaurant system
- Healthcare system
- Netflix as a company and its architecture
- **Bitcoin**

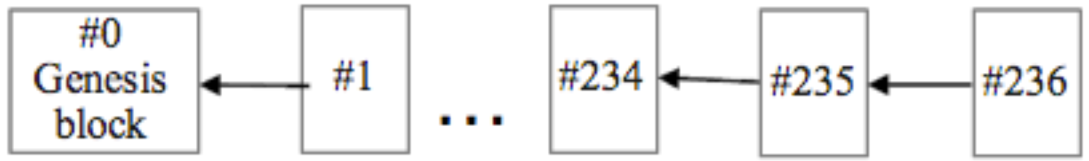
Industries are in interested in Bitcoin

- Banks
- Music
- Retail
- Supply Chain
- Manufacturing

- But they see **issues** with the Bitcoin protocol, so they are investing in **Blockchain**

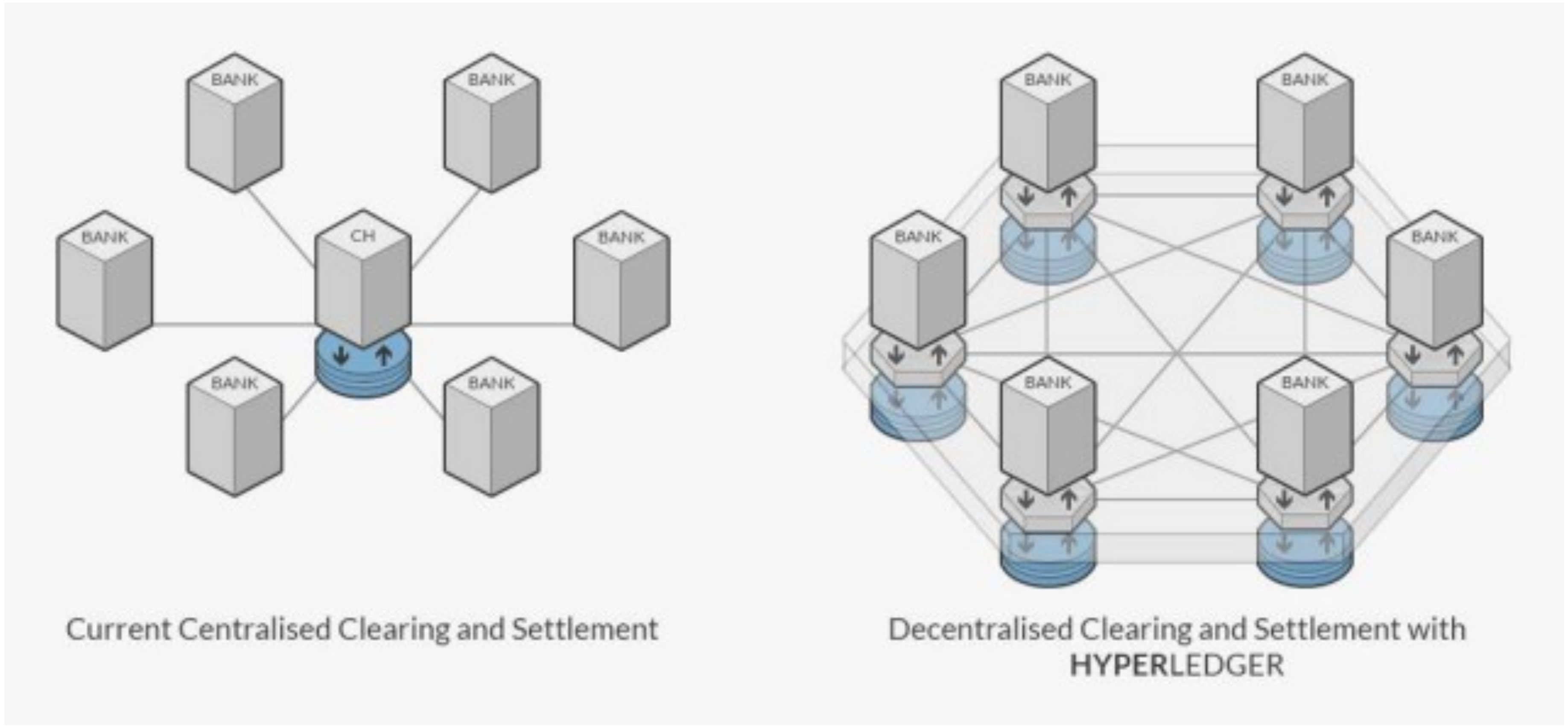
What is a Blockchain ?

- **A chain (sequence) of <u>blocks</u> of transactions**
  - Each block consists of a number of transactions



- **A Blockchain is just a Distributed Database with:**
  - Added security
  - Consensus
  - Data immutability
  - Smart Contracts (chaincode)
  - Authorization & Authentication
  - Record keeping

Traditional vs. Blockchain


BANK
BANK
BANK
CH
BANK
BANK
BANK
Current Centralised Clearing and Settlement
BANK
BANK
BANK
BANK
BANK
BANK
Decentralised Clearing and Settlement with
HYPERLEDGER

	Proof of Work (Bitcoin, Ethereum)	State machine replication (Hyperledger, Corda)
Membership	Permissionless	Permissioned
User IDs	Decentralized, Anonymous (Decentralized protection by PoW compute/hash power)	Centralized, all Nodes know all other Nodes (Centralized IDM protects against Sybil attacks)
Scalability (no. of Nodes)	Excellent, >100k Nodes	Verified up to few tens (or so) Nodes
Throughput	7 tx /sec upper bound (Bitcoin)	>10k tx /sec with existing implementations in software
Power efficiency	>1 GW (Bitcoin)	Good (commodity hardware)
Forks in blockchain	Possible (leads to double spending attacks)	Not possible
Consensus	No	Yes, with BFT protocols
Cryptocurrency	Yes	No
Anti-Fragile	Yes	???

Problem

- Important institutions are rushing to implement Blockchain.
- Most implementations are untested and will likely have bugs.
- Hypothesis: Removing key elements from anti-fragile system like Bitcoin will make it more fragile
- What can I use to test Distributed Systems like Blockchains?
  - Formal methods
  - **Failure injection**
  - Instrumentation

Related Work

- Random fault injection (Basiri et al., 2016)
- Byzantine fault injection (Martins et al., 2013)
- Lineage-driven fault injection (Alvaro et al., 2015)

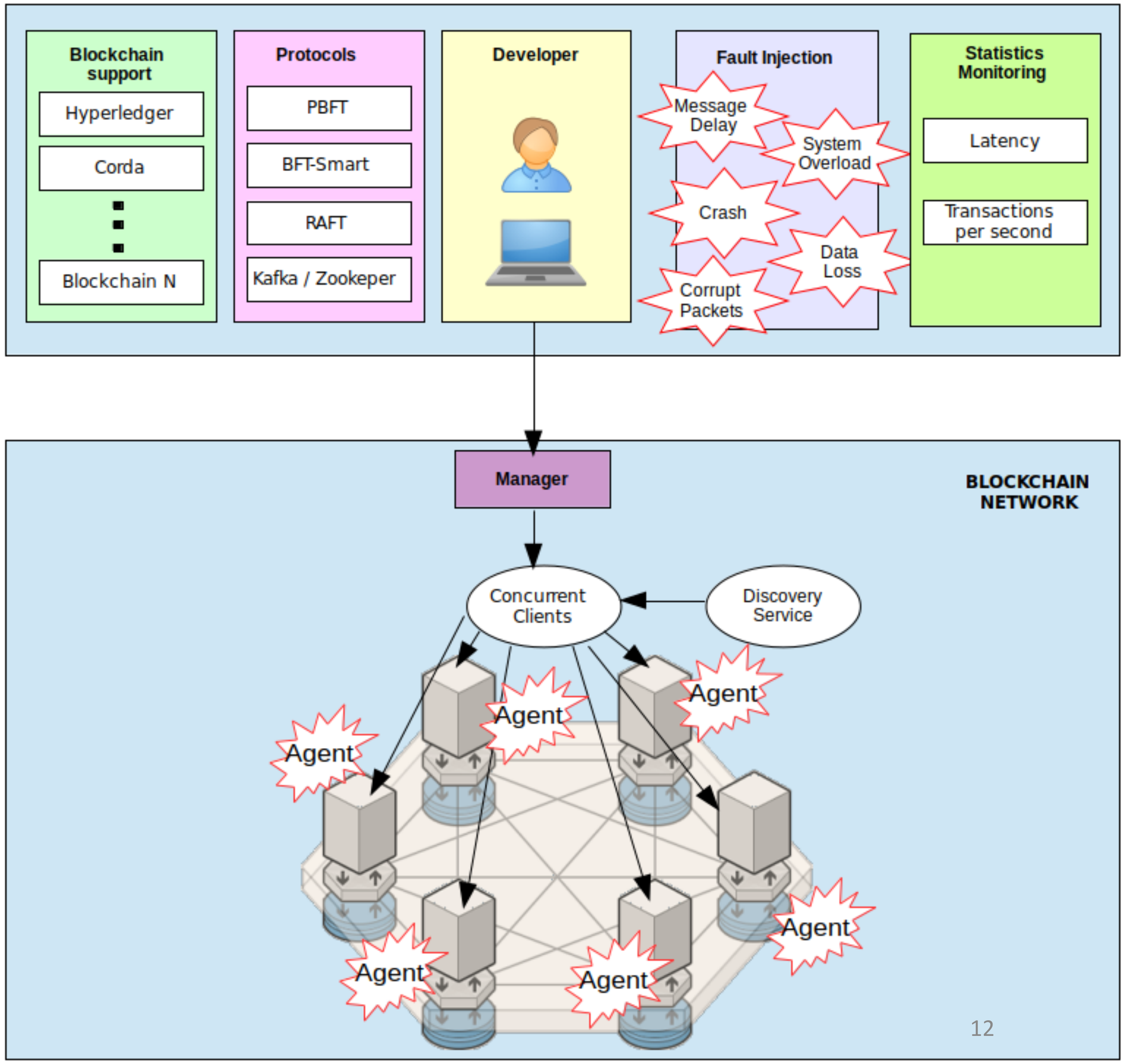
Motivation

- Blockchain technologies do not have any fault injection frameworks
- Blockchain tech hasn’t been formally verified
- Available fault-injection solutions do not cover Byzantine failures

My proposal

- Systematic Byzantine Failure Injection for Blockchain technology.
- Byzantine faults in consideration
  - Crash
  - Message Delay
  - Corrupt Packets
  - System Overloading
- Systematic and recoverable injection
  - Fault Type
  - Fault Parameters (fault duration, delay time, # clients)
- Blockchain and protocols in consideration
  - Hyperledger Fabric - PBFT & Kafka
  - Corda - BFT-Smart & RAFT

Architecture for Failure Injection



Challenge

- Create a general solution to inject Byzantine failures into Blockchains despite their differences:
    - Language
    - Architecture
    - Protocol

Next Steps

- Complete implementation of the Failure injector
- Perform large scale experiments with Hyperledger Fabric and Corda
- Introduce Smart failure injection like LDFI
- Start the base line for a Benchmark for permissioned Blockchain

END

- Problem: Recent interest in Blockchain technologies that remain untested. They lack good testing framework to verify and compare them. We don’t know if they are anti-fragile.

- Contribution: Systematic Byzantine Failure Injection for Blockchain technology. Aims to help benchmark Blockchain technologies and endow them with anti-fragility if used in production.

Example of Failure

- http POST http://injector:8080 cmd="WAIT" period=500 type="DELAY"

- http POST http://injector:8080 cmd="START" type="DELETE" path="/var/lib/hyperledger/data"

- http POST http://injector:8080 cmd="WAIT" type="DOWN" iface="eth0"