Final Presentation
Praktikum: Cloud Databases

Group 10

Beyza Altuntas
Ibrahim Kaplan
Halil Sahiner

Outline

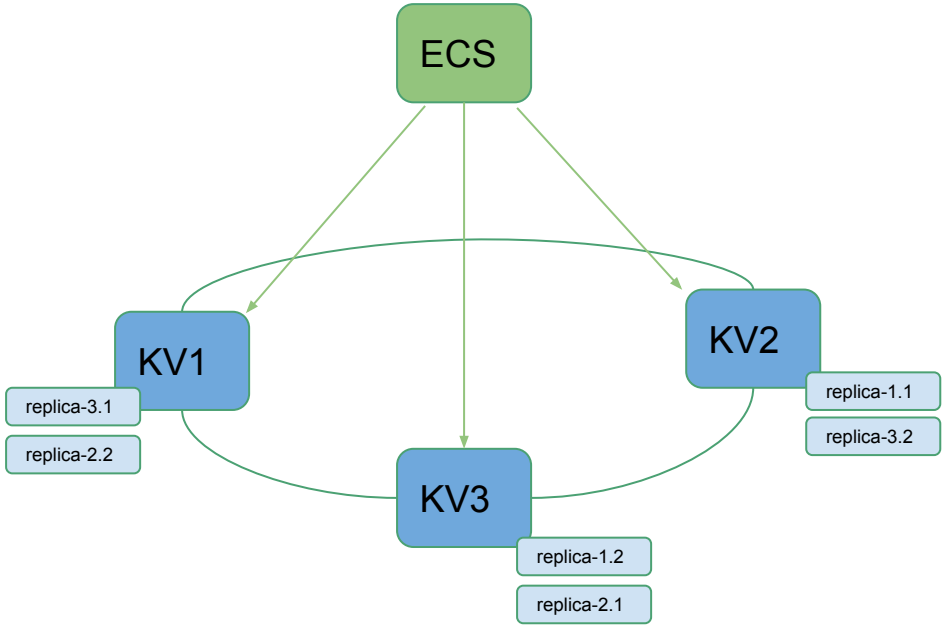
1. Motivation
2. Contributions to the Architecture
3. Election Mechanisms
4. Performance Evaluation
5. Conclusion

Cloud Systems

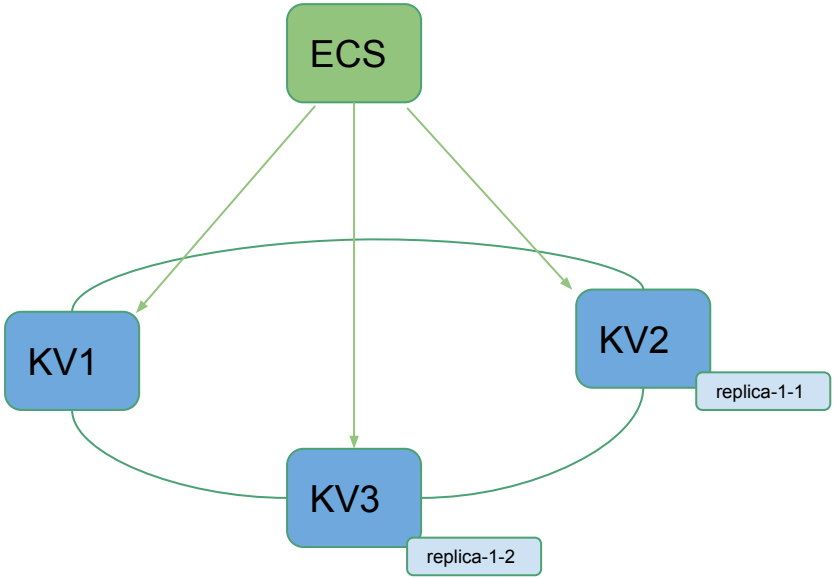
Cloud Databases

- Features
  - Distributed
  - Replicated
  - Scalable
  - ***Fault-tolerant***
- Why?
  - Cloud systems uses and utilizes commodity hardware, the whole ecosystem is build around the flexible, fault-tolerant architectures.
- We have fault-tolerance for KV-Servers, but what about ECS?
  - Single point of failure

After Milestone 4

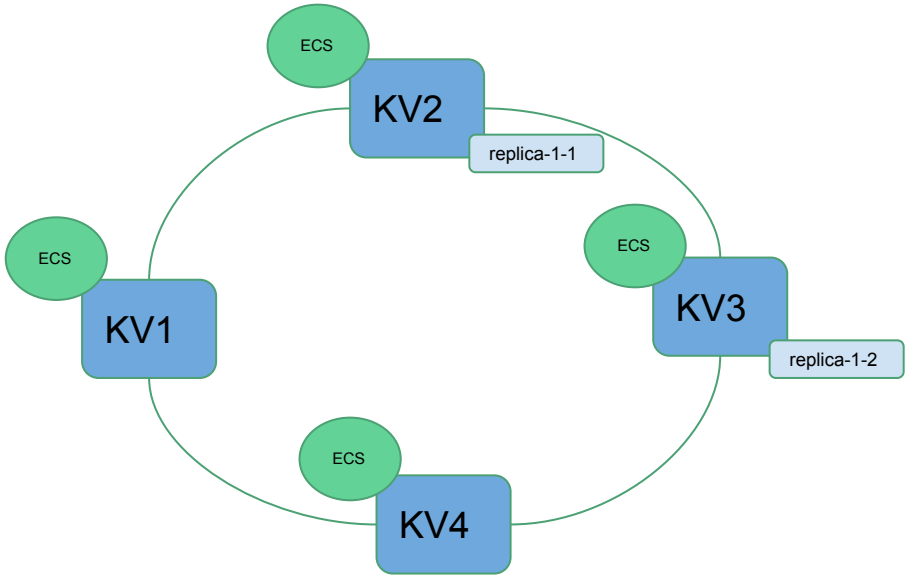


After Milestone 4



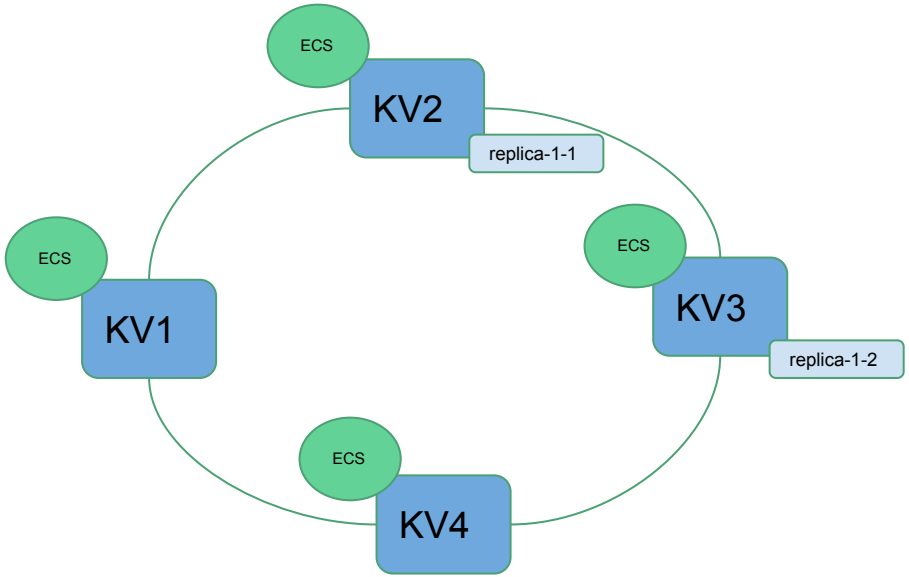
Seems more like a client-server architecture?

Distributed, replicated, scalable, fault-tolerant



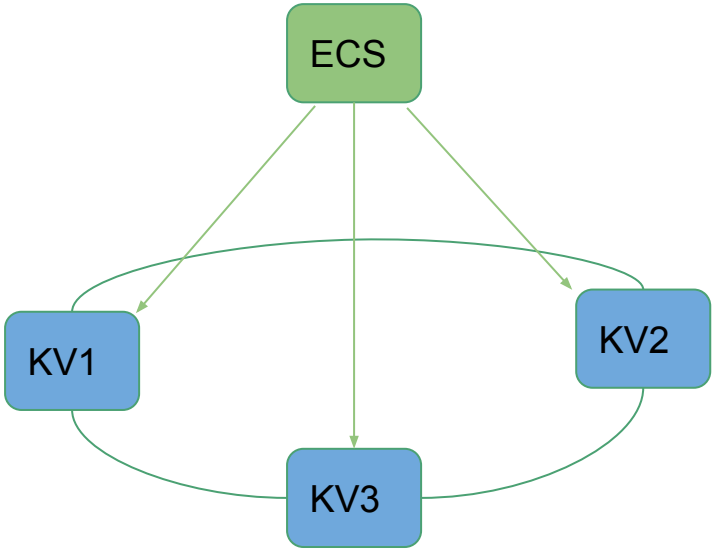
P2P system

Distributed, replicated, scalable, fault-tolerant

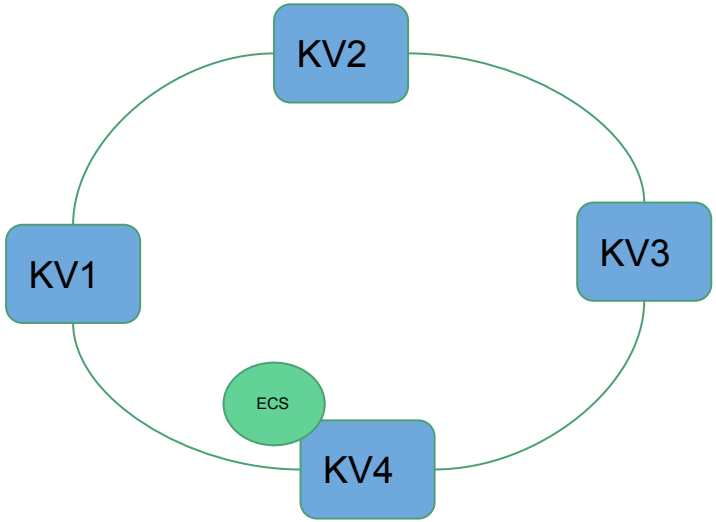


How?

ECS Responsibilities

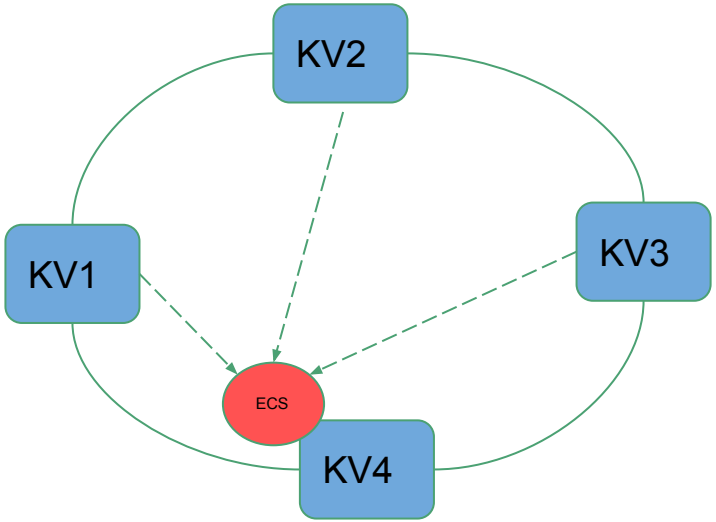


One central ECS server, works indefinitely



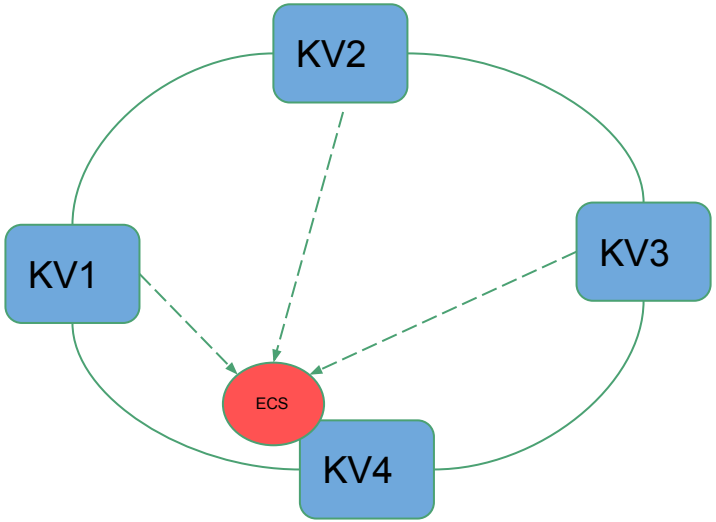
Elected KV-Server is responsible for ECS - *yey!*

In case of failure



Whenever a KV-Server detects the failure of ECS, it starts an ***election***

In case of failure

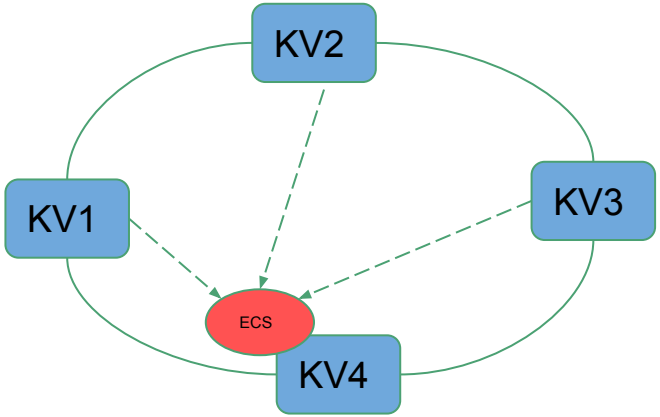


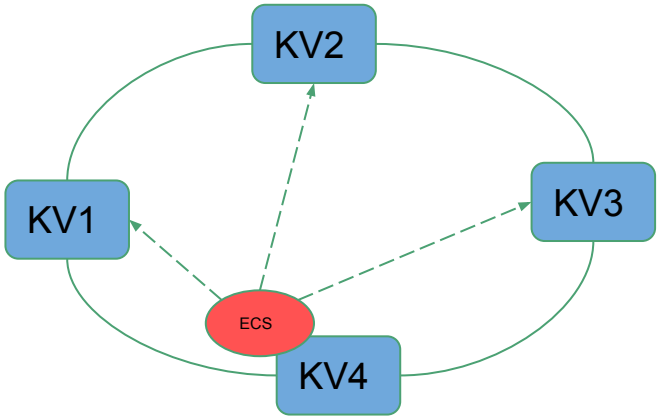
How can they detect?

Whenever a KV-Server detects the failure of ECS, it starts an ***election***

Two-sided heartbeat mechanism

- In the last milestone, we had a centralized ECS Server that regularly checks the status of the KVServers
- Now, KV-Servers also checks the status of the node who has both KVServer and ECS capabilities





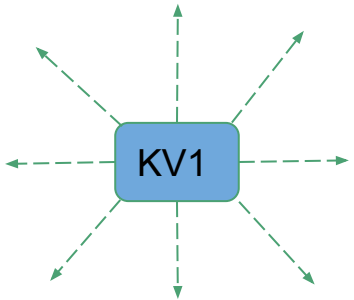
Leader Election

- 2 Cases
  - On start
  - When the existing leader fails - *long live the king!*

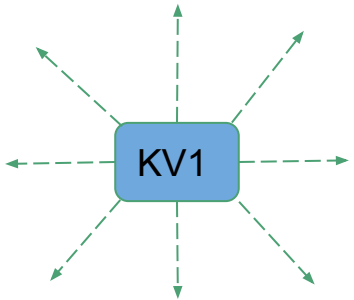
Election mechanism - on start

- Searches for other servers in a specified IP range
  - Ex. 127.0.0.0-127.0.0.100
- If it gets a reply,
  - it saves the location of the ECS, it initiates the communication, retrieves the *keyrange*.
- If it doesn’t get a reply back *any* of them,
  - it declares itself as an ECS server.
  - Sounds like ***Bully Election*** from Distributed Systems?
- Drawback: Longer cold start

Election mechanism - on start

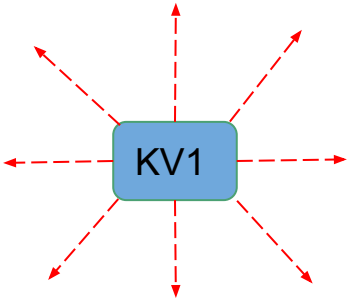


Election mechanism - on start

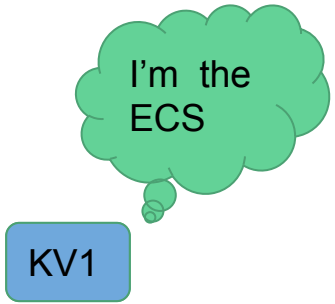


finite set : configurable
IP range

Election mechanism - on start



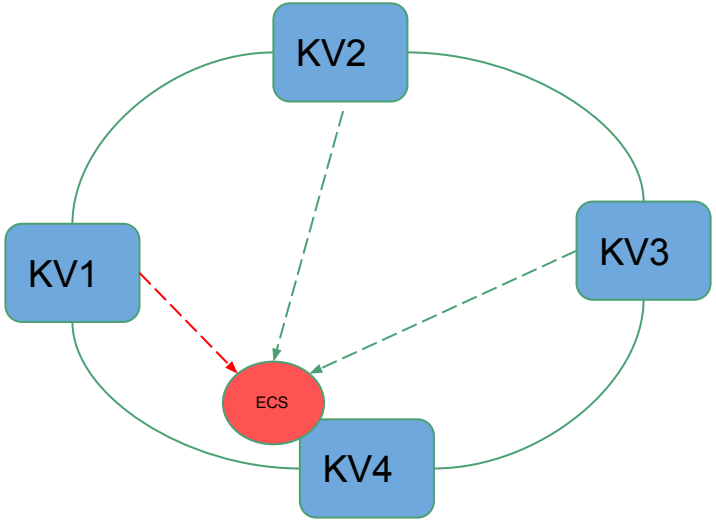
Election mechanism - on start



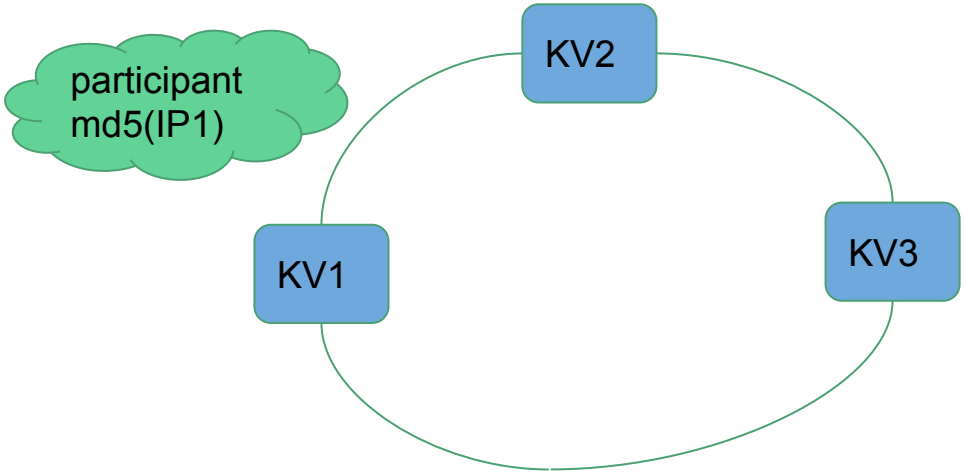
Election mechanism - during failure

- When a KV-Server detects the failure of ECS, it starts an election by sending participant message to its successor.
  - ***participant md5(server_ip)***
- If a KV-Server receives a participant message,
  - Received Hash Value > Server's Hash Value => passes received hash value to its successor
  - Received Hash Value < Server's Hash Value => passes own hash value to its successor
  - Received Hash Value = Server's Hash Value => declares itself as leader by ***elected md5(server_ip)***
- When a KV-Server receives elected message,
  - It updates ECS server address on its storage
  - If the hash value is not equal to its own, it passes the message to its successor
- Full implementation of *Chang - Roberts* ring based election algorithm.

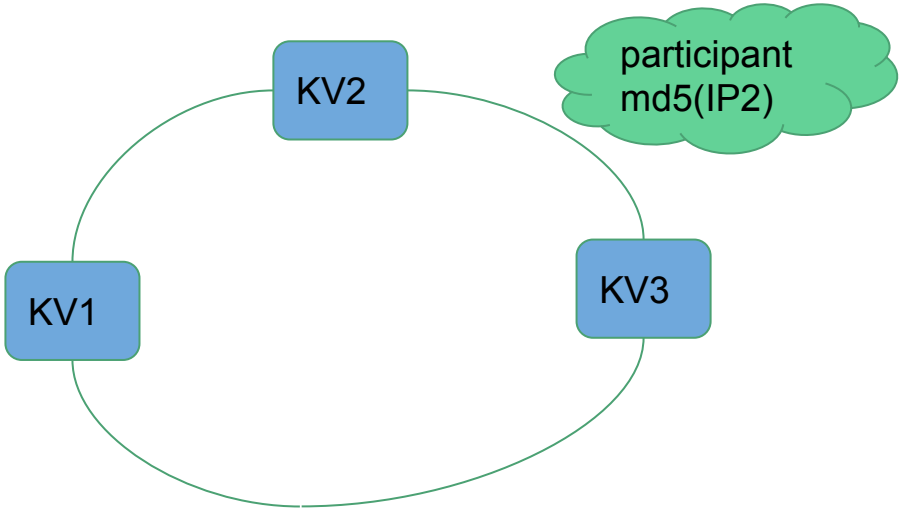
Election mechanism - during failure



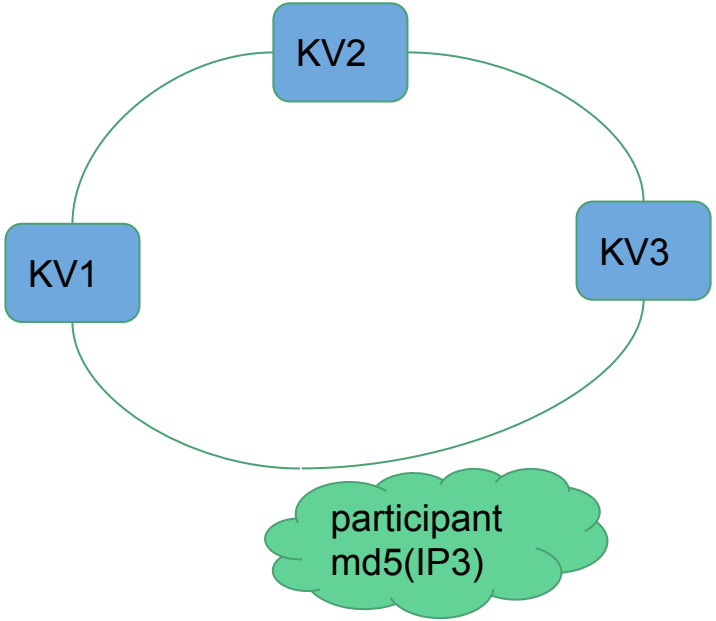
Election mechanism - during failure



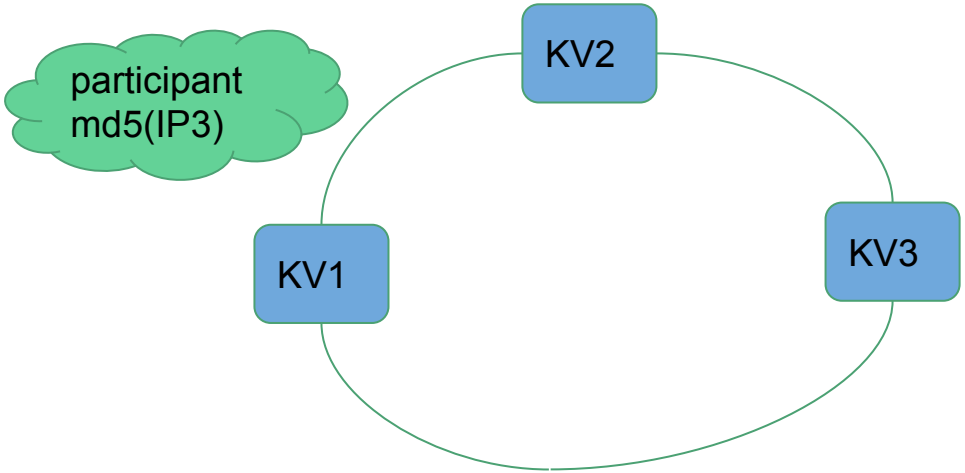
Election mechanism - during failure



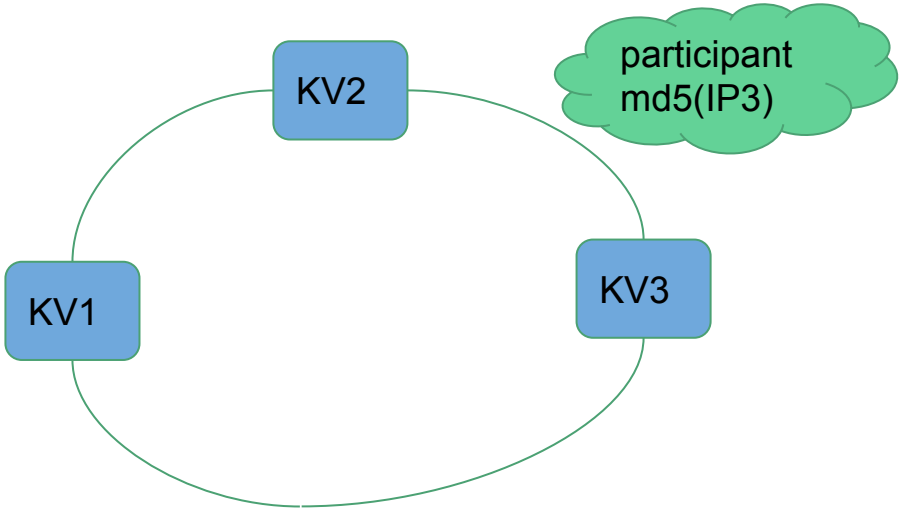
Election mechanism - during failure



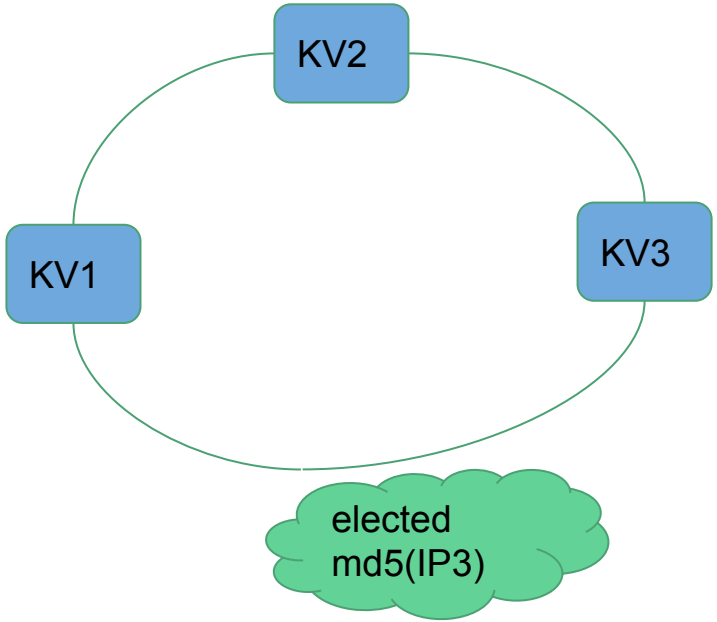
Election mechanism - during failure



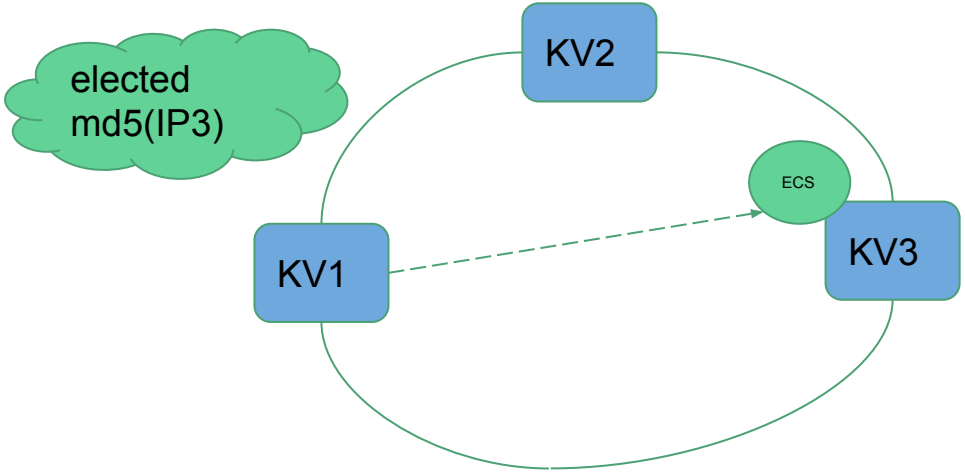
Election mechanism - during failure



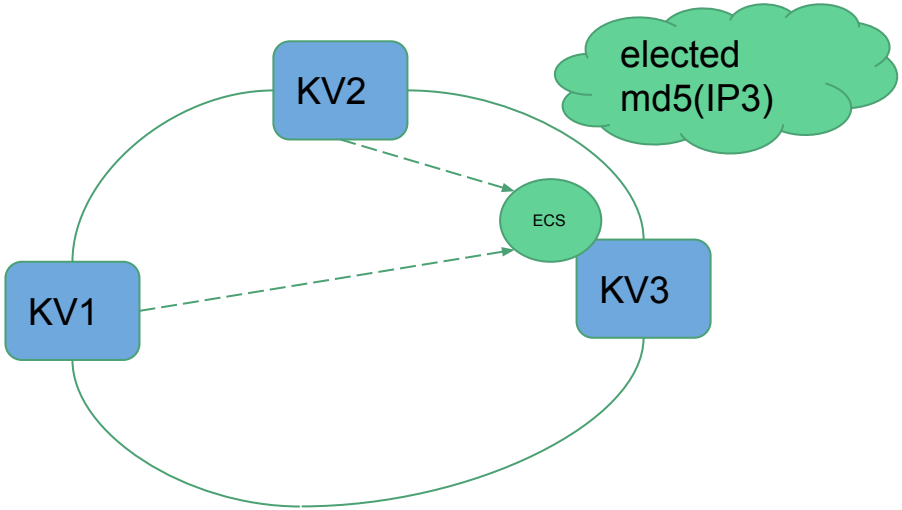
Election mechanism - during failure



Election mechanism - during failure



Election mechanism - during failure



Election finishes

Performance Evaluation

Performance Evaluation

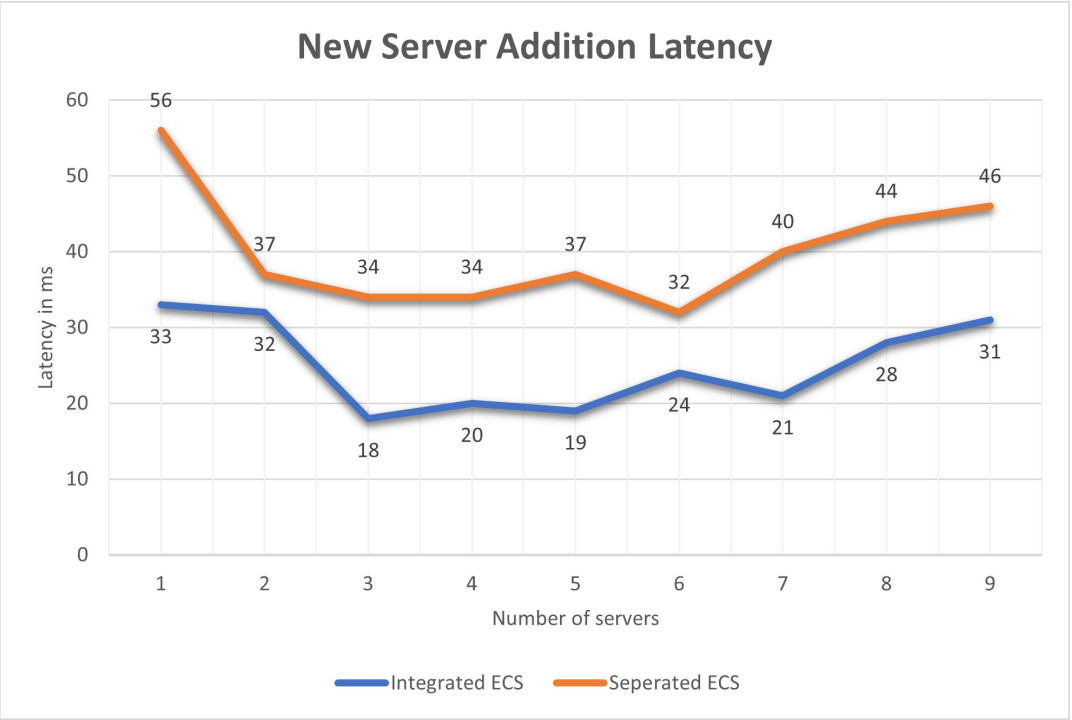
1. Server Addition to Ring with Election
2. Server Addition to Ring with Client Communication
3. ECS Election Latency
4. Write Latency & Throughput

Machine we tested on:

CPU: AMD Ryzen 5 5600X 6-Core Processor
RAM: 8X4 GB 3000 MHz

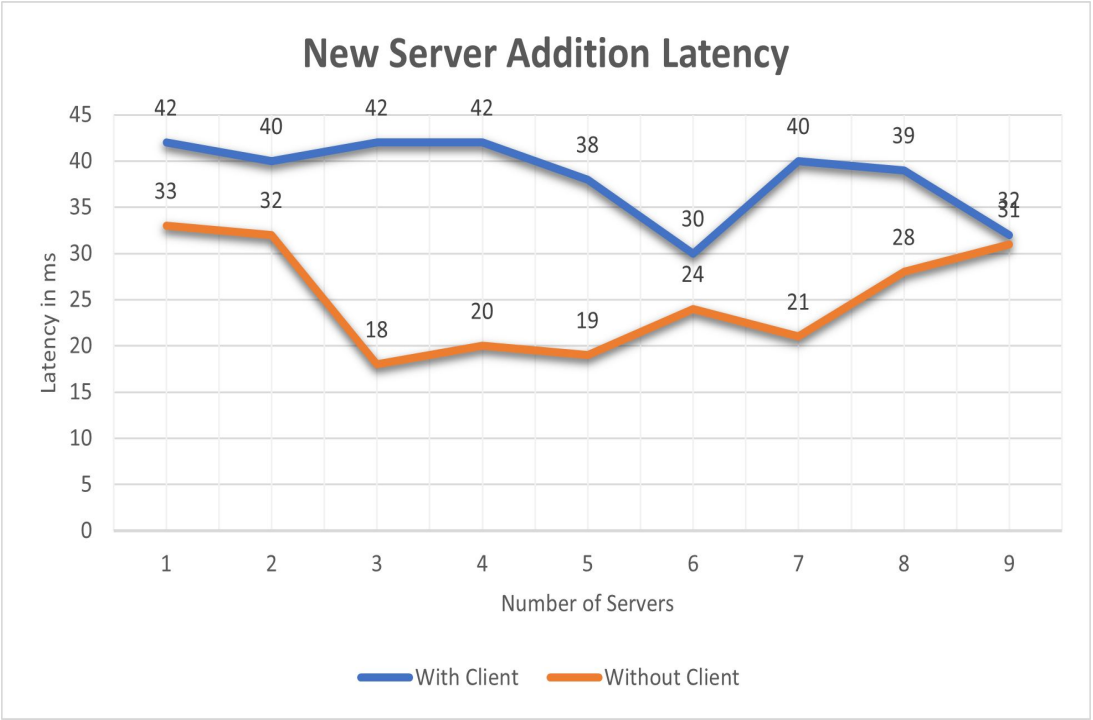
Server Addition to Ring with Election

- Without data in servers
- Purely communication overhead



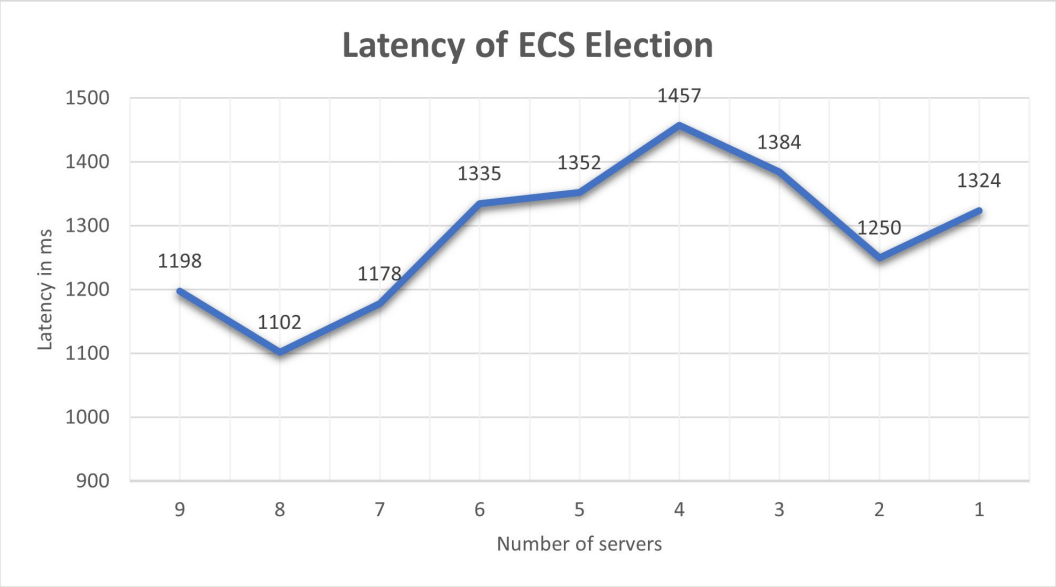
Server Addition to Ring with Client Communication

- Effect of one client that writes data

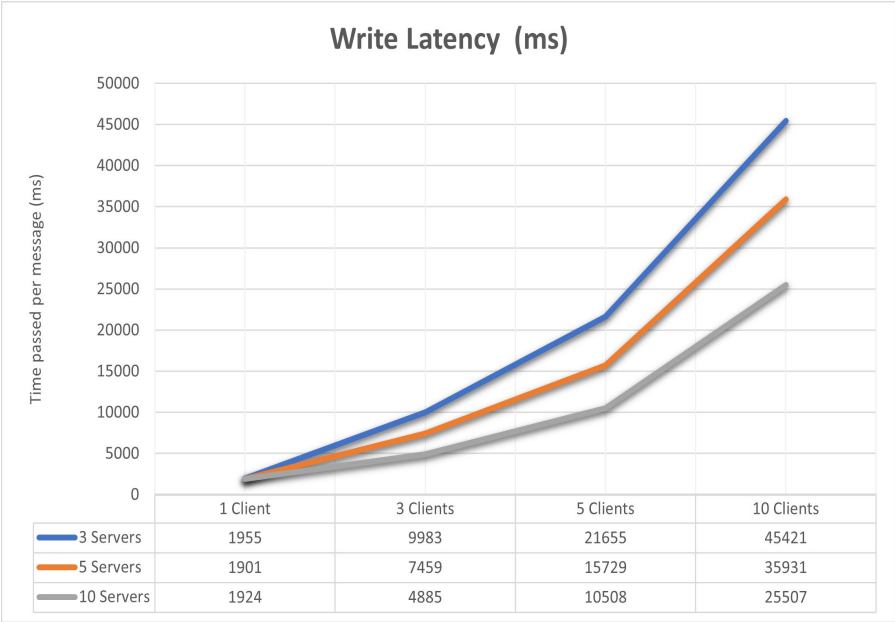


ECS Election Latency

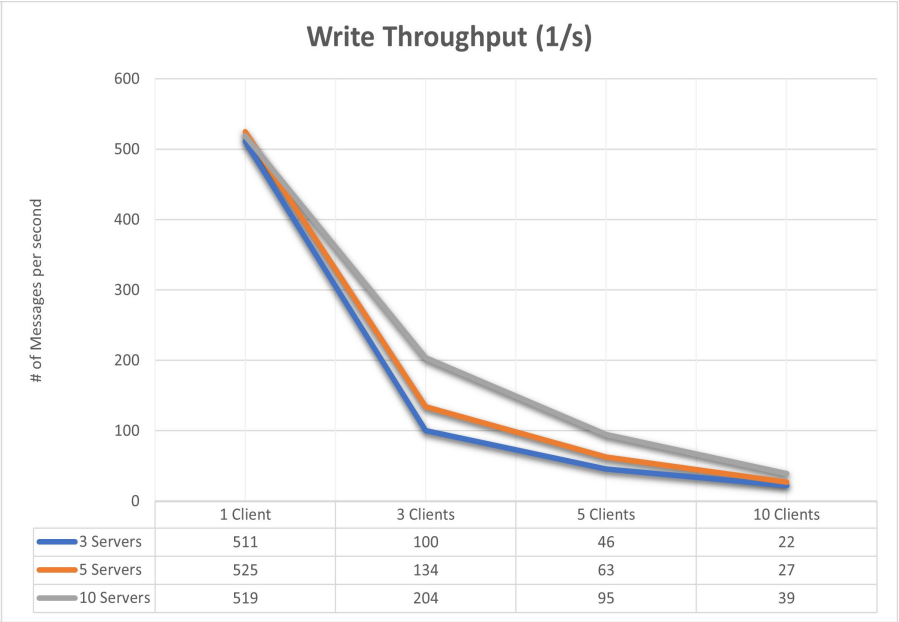
- The failure detection interval: 700 ms
- Worst Case Communication Overhead : 3n-1



Write Latency & Throughput



	1 Client	3 Clients	5 Clients	10 Clients
3 Servers	1955	9983	21655	45421
5 Servers	1901	7459	15729	35931
10 Servers	1924	4885	10508	25507



	1 Client	3 Clients	5 Clients	10 Clients
3 Servers	511	100	46	22
5 Servers	525	134	63	27
10 Servers	519	204	95	39

Conclusion

Pros:

- More flexible system in case of node failures
  - No more single point of failure
  - Less hardware costs -*no need for powerful and robust leader node*-
- Fault-tolerance for ***every*** node in the system

Cons:

- More network overhead between nodes - *technical debt*
  - KVServers check ECS in case of failures => election start
    - Two sided heartbeat mechanism
  - Election messages
- Assumptions of ring based election system

Thank you for listening.

Questions?