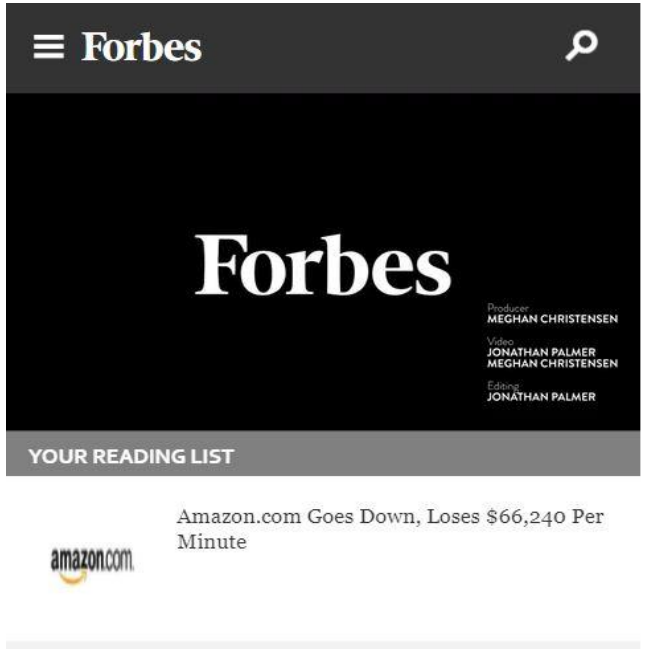
Dynamo: Amazon's Highly available key-value store

PRESENTED BY: MEGHNA GARG

CMPT 843



Tech

AUG 19, 2013 @ 03:59 PM 24,045

2 Free Issues of F

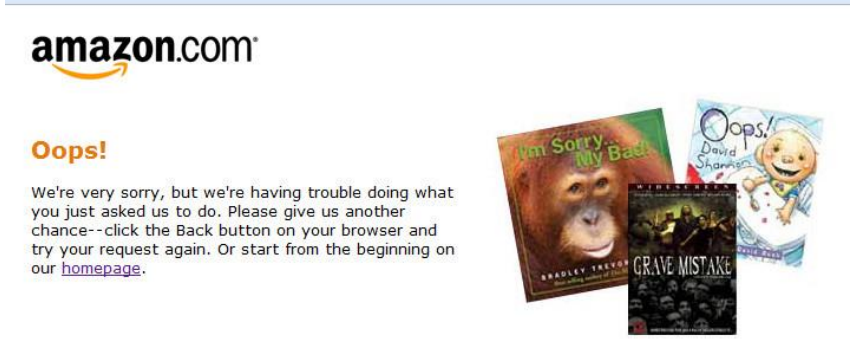
Amazon.com Goes Down, Loses $66,240 Per Minute

Ad closed by Google

Kelly Clay, CONTRIBUTOR

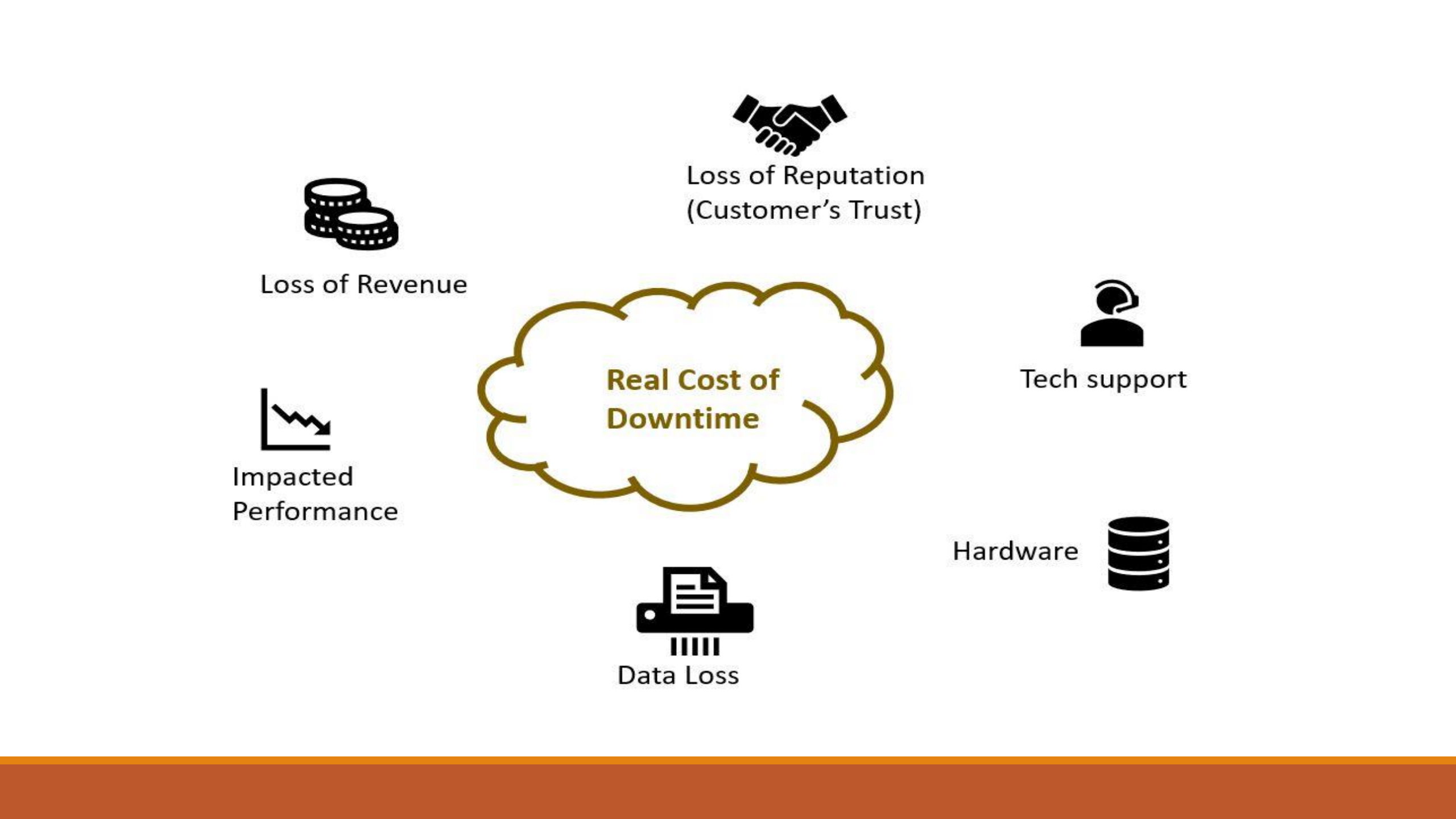
I write about social media, startups and technology trends. FULL BIO

Opinions expressed by Forbes Contributors are their own.



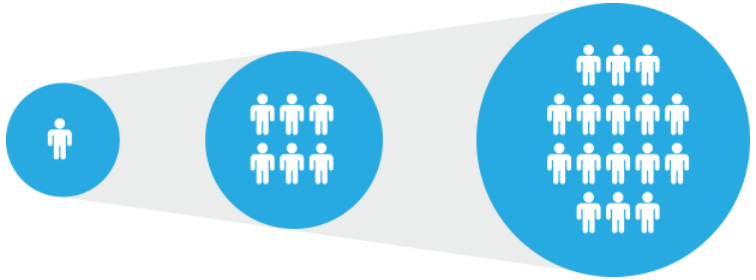
Total Outage time : Approximately 30 minutes

Total Loss incurred : Around $2 million


Loss of Reputation
(Customer's Trust)
Loss of Revenue
Real Cost of
Downtime
Tech support
Impacted
Performance
Hardware
Data Loss

Driving Forces

Reliability



Scalability

Availability

Roadmap

- Design Considerations
- What makes Dynamo different?
- System architecture
- Performance
- Industrial Use Case
- Conclusion

Design Considerations

- Trade-off between availability and consistency : No Compromise with correctness
- Optimistic replication techniques : Conflicting changes are detected and resolved.
- Privileged for writes : Rejecting customer updates could result in a poor customer experience.
- Incremental Scalability : Scale with minimal impact on system.
- Symmetry : Equally responsible nodes , supports decentralization
- Heterogeneity : Proportional work distribution.

What makes Dynamo Different?

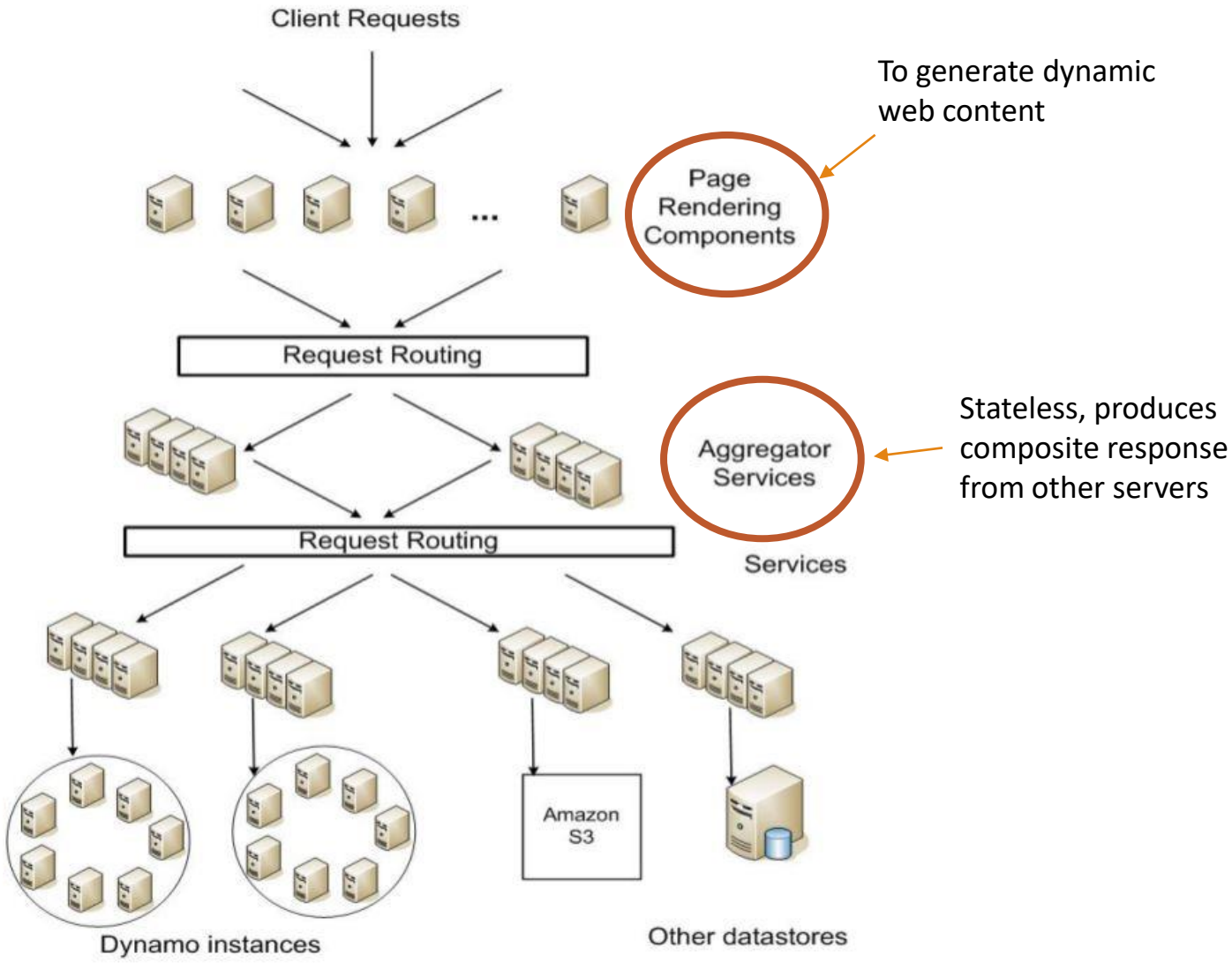
Always Writable : No updates rejected due to failures

Single administrative domain with **trusted nodes.**

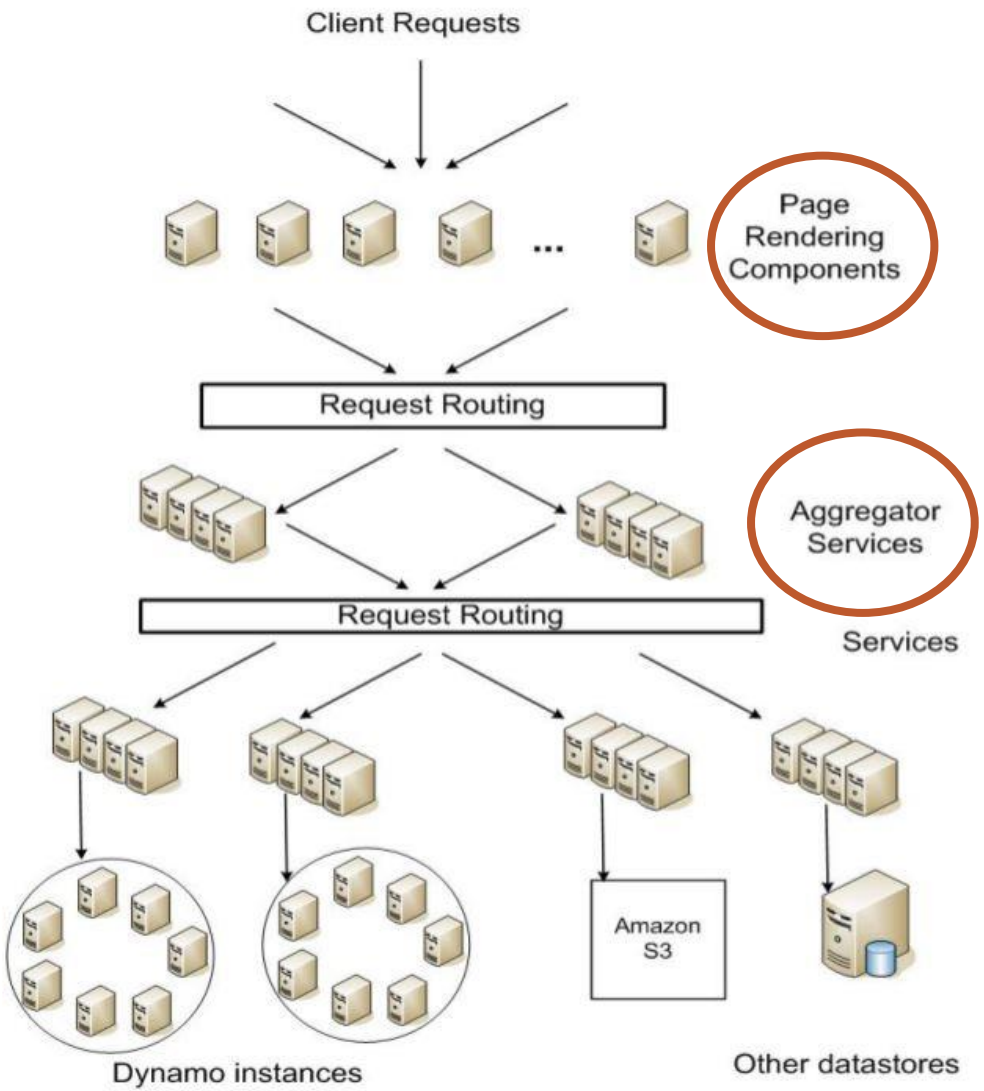
Unlike traditional databases, **it does not require support for complex relational schema**

Latency sensitive application oriented where 99.9% or R/W performed within few milliseconds

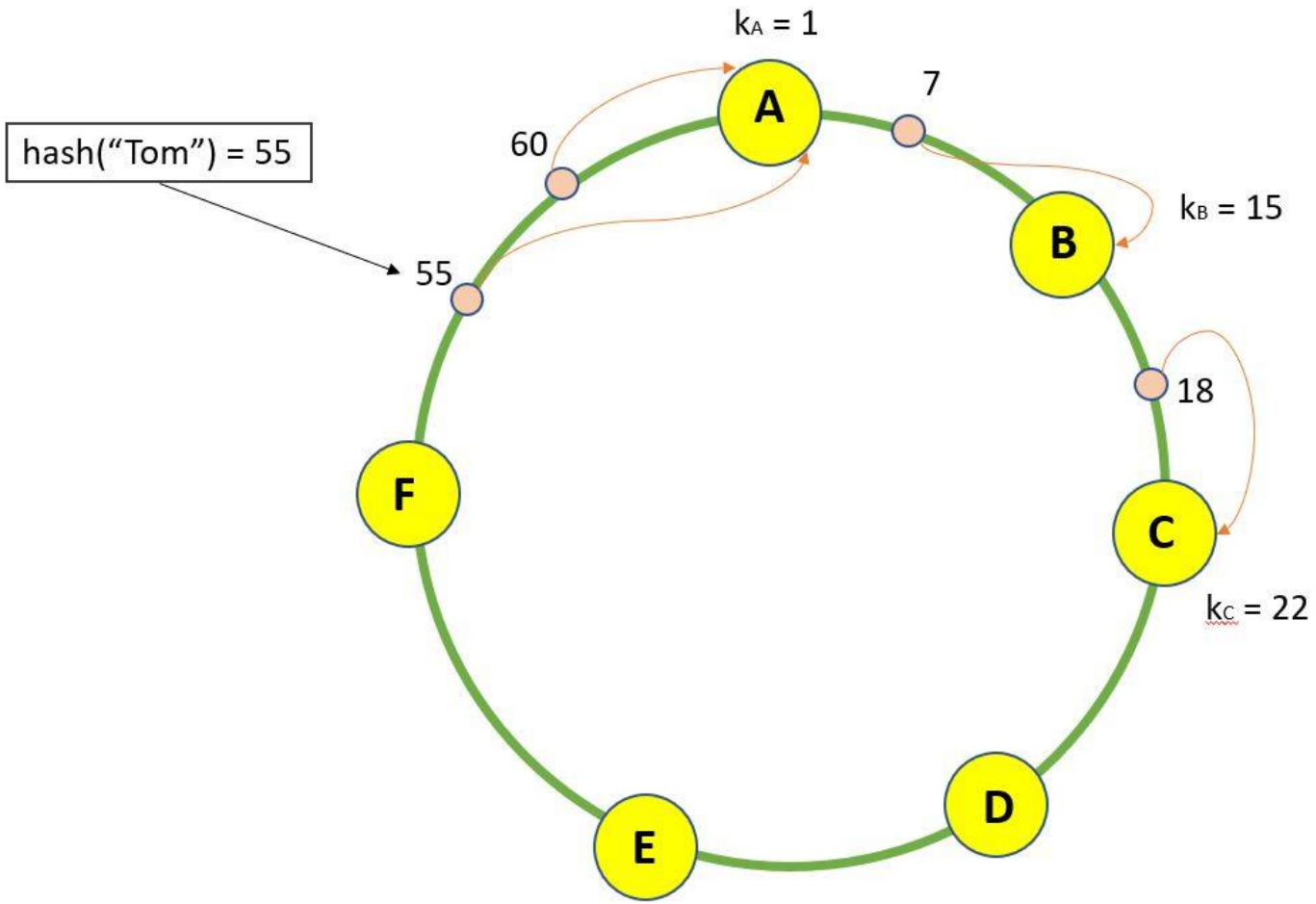
Service-oriented Architecture



Service-oriented Architecture



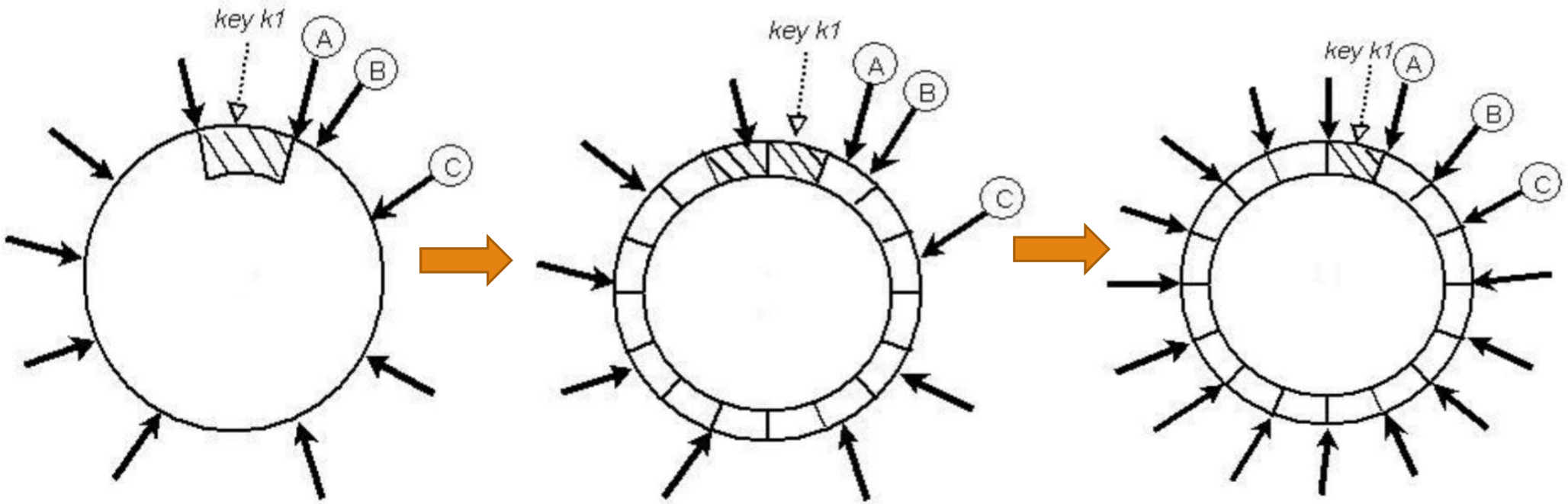
- A Page request may involve over 150 services , with multiple dependencies.
- SLAs (Service level Agreements) : Clients and services agree on an expected latency and request rate distribution.
- At Amazon, SLAs are defined to target **all** customers (99.9 percentile)
- Example: Guarantee a response within 300ms for 99.9% of its request, for a peak client load of 500/sec.



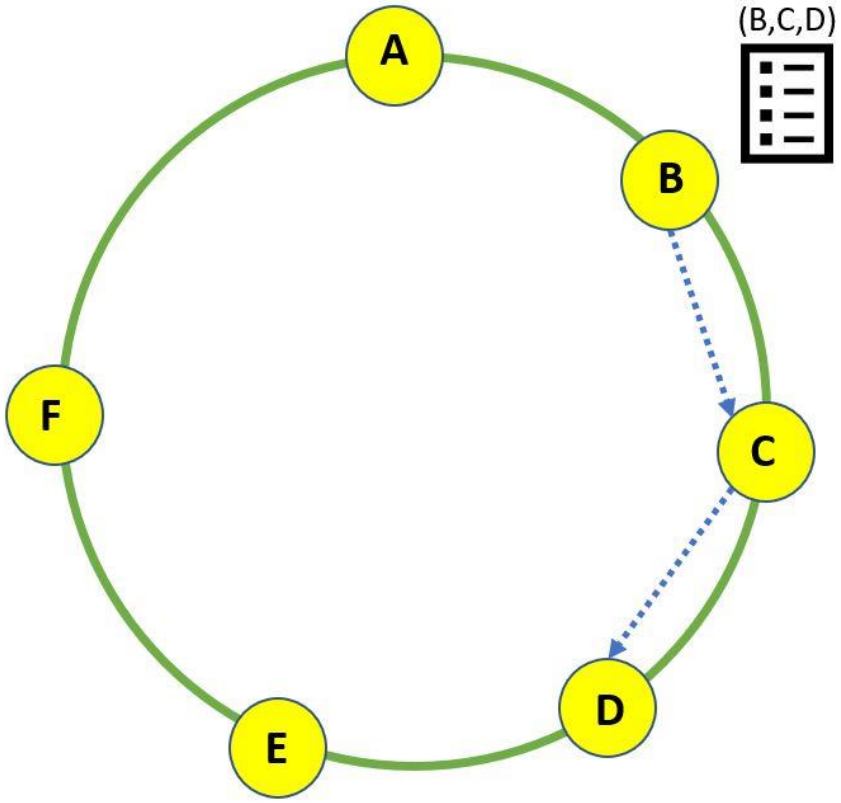
Partitioning

- Hash Function: Outputs random numbers ranged in circular space. Ex : Modulo
- Walk clockwise to find the first node, with position > item’s position.
- Benefit: Departure/Arrival effects immediate node.
- Drawback : Skewed distribution

Uniform Load/Distribution



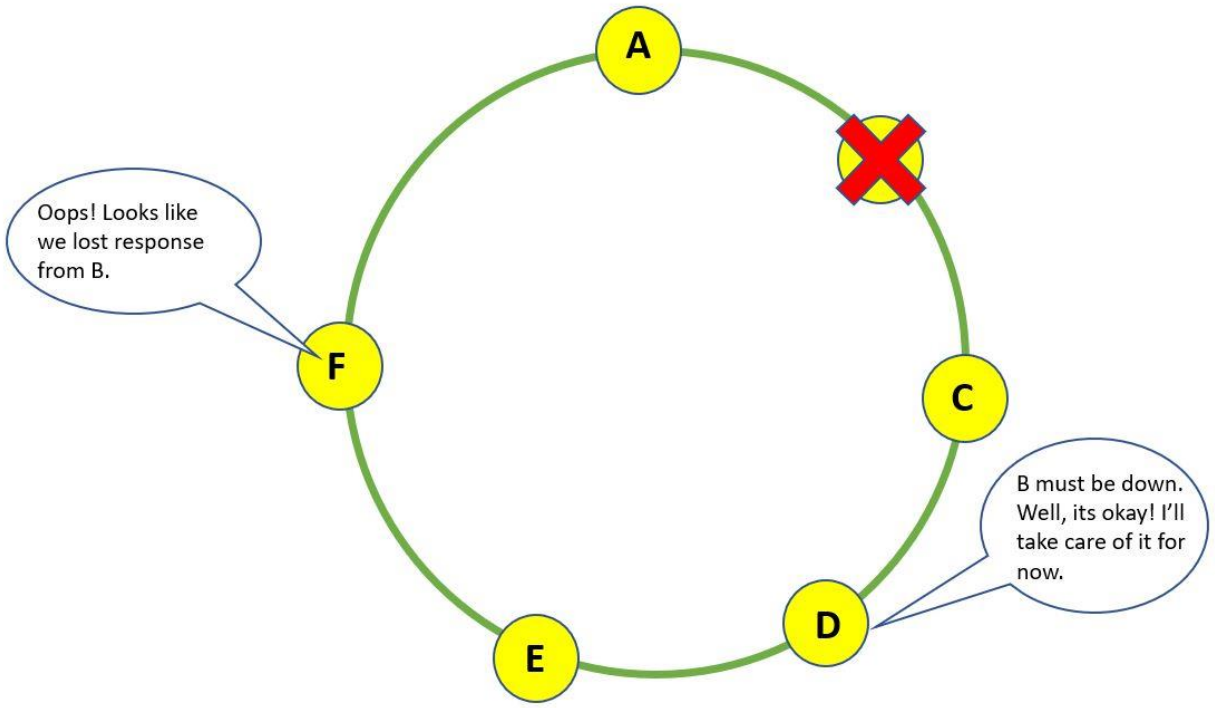
	Strategy 1	Strategy 2	Strategy 3
Keys per node	Random	Random	Q/S (Q>>N, S = No. of nodes in system)
Partition size	Unequal	Equal sized , Q	Equal sized , Q
Description	Required recalculation of key range on node arrival, bootstrapping	Decoupling of data partition and placement	Fast bootstrapping and recovery



Replication

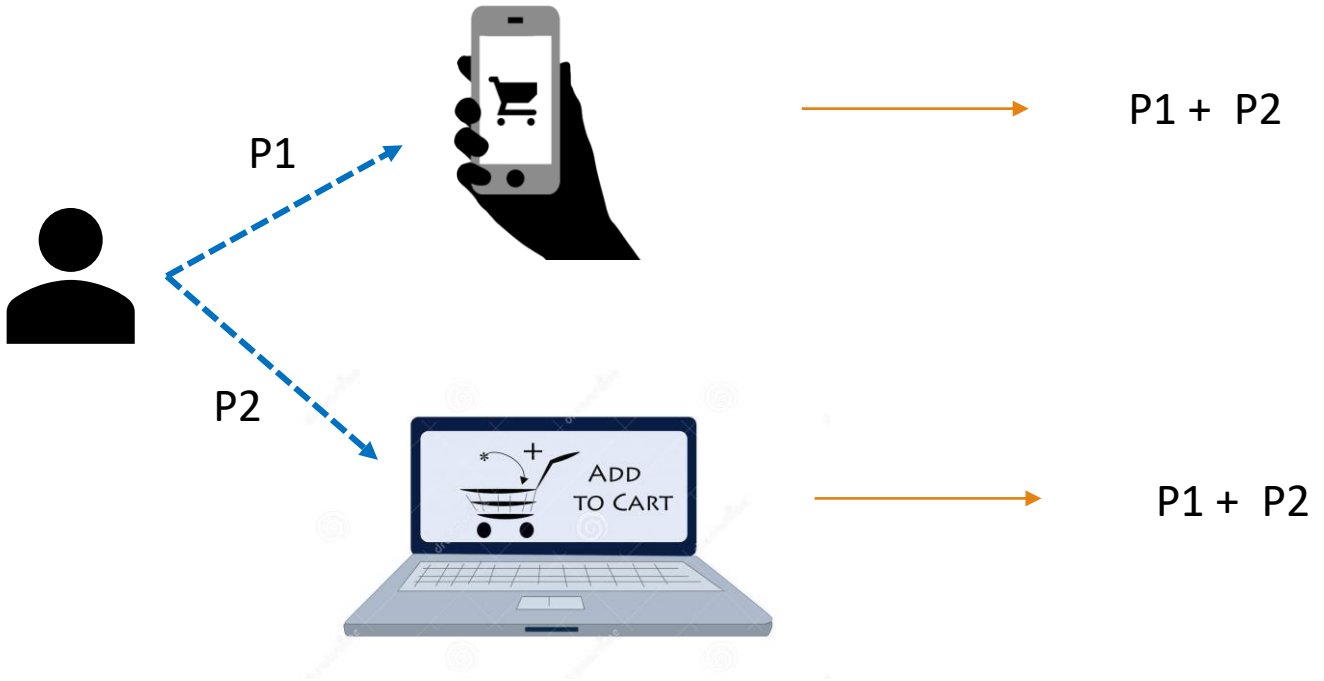
- Each node maintains a preference list.
- Preference list : Contains the nodes which it is responsible for.
- Each key is assigned to a coordinator.
- Coordinator stores data locally and on N-1 successors.
- Ex: D is responsible for B and C for N=3 . D stores keys that fall in range (A,B],(B,C],(C,D].

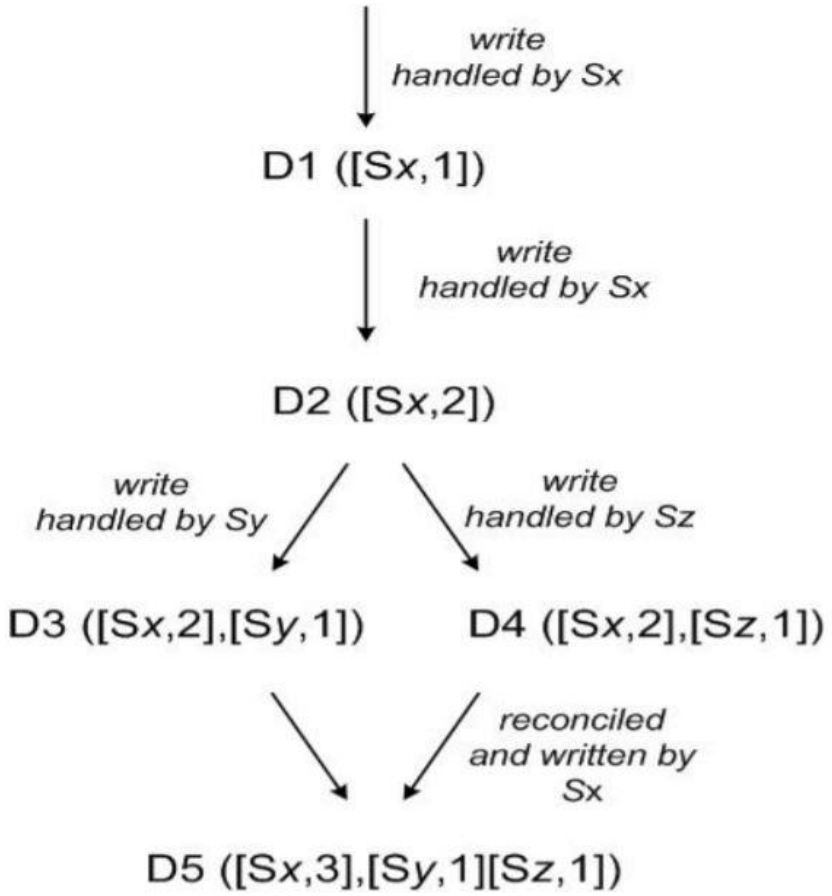
Advantages



- Node Crashes: it can be replaced by other nodes in preference list.
- Node added : adjust only its N-1 successors.
- Flexibility: N can be decided based on the capacity.

Data Versioning

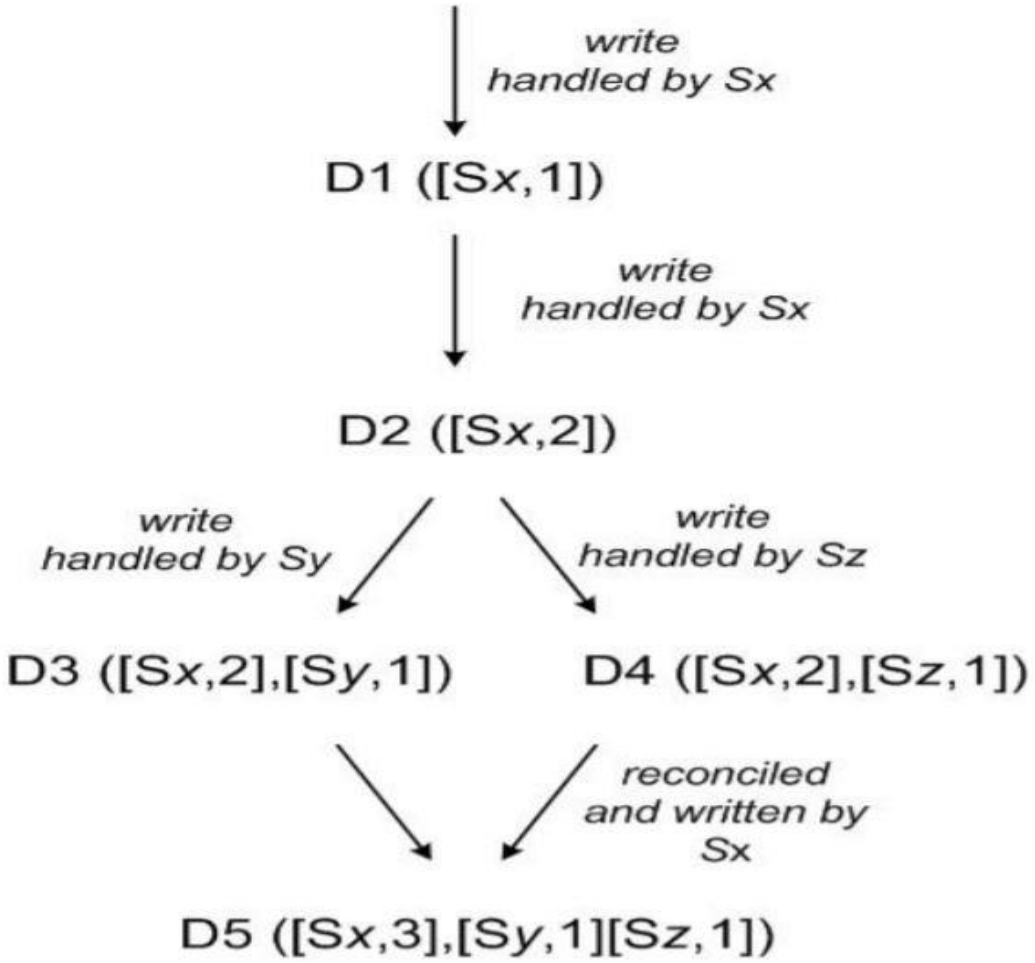




Data Versioning

- Each modification by the user is considered new and immutable by Dynamo
- Allows multiple version of an object to be present at the same time
- For conflicting versions of an object, performs collapse operation.
- Version Clocks : list of (node, counter) pairs associated with each version of object.

Data Versioning



ISSUE

The size of vector clocks grow if many servers are involved for writes

Handling Failures

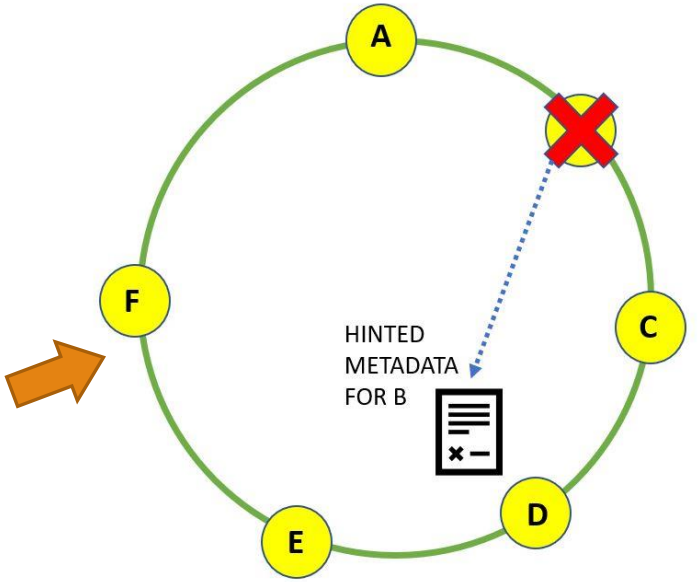
Failures

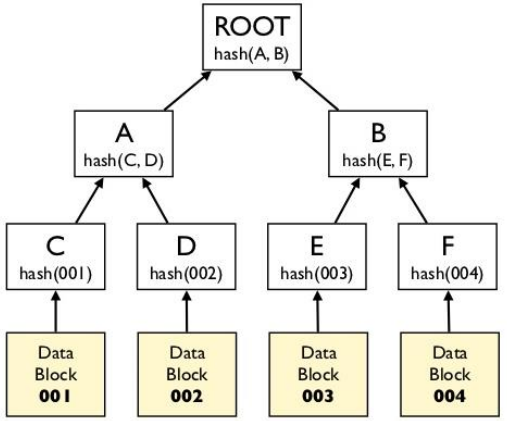
Temporary

Hinted Handoff

Permanent

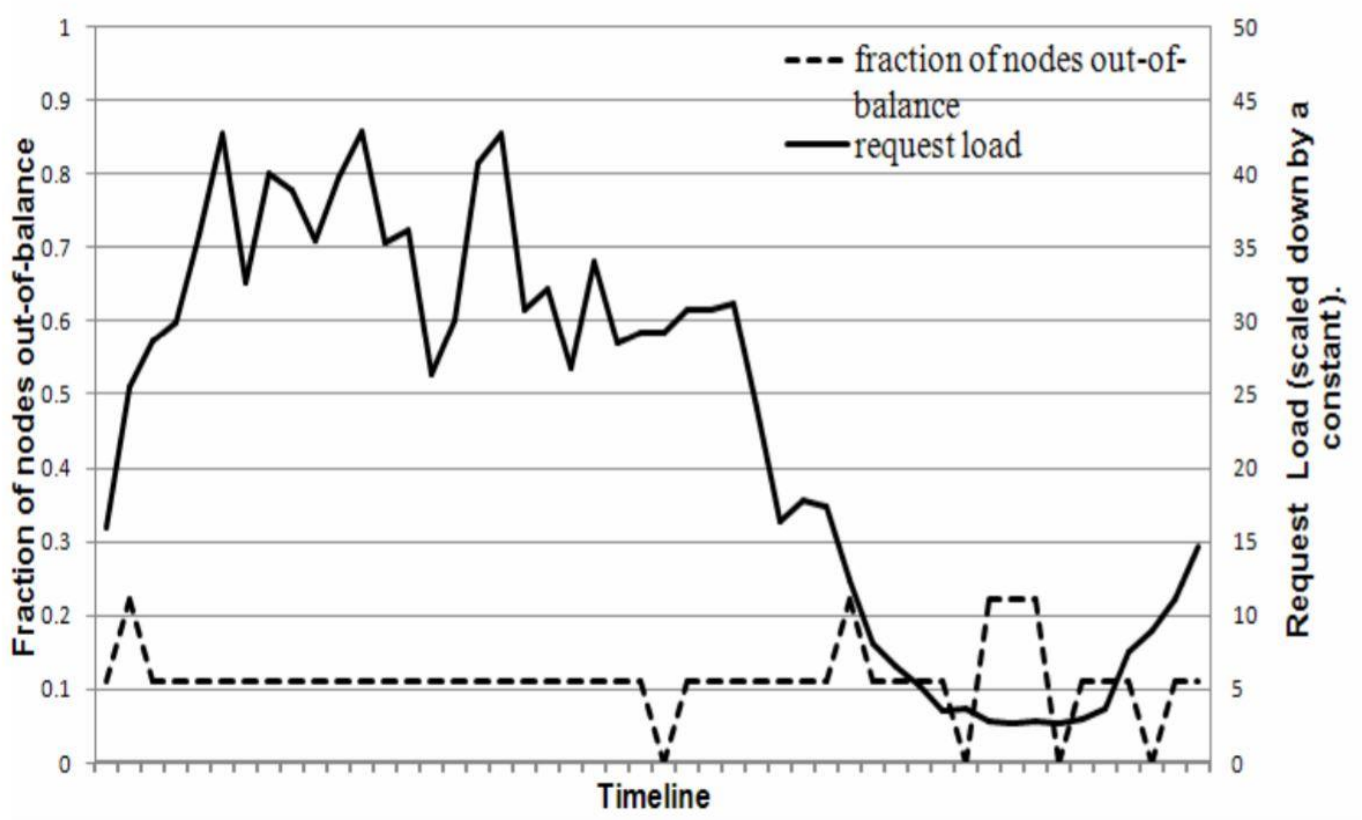
Replica Synchronization





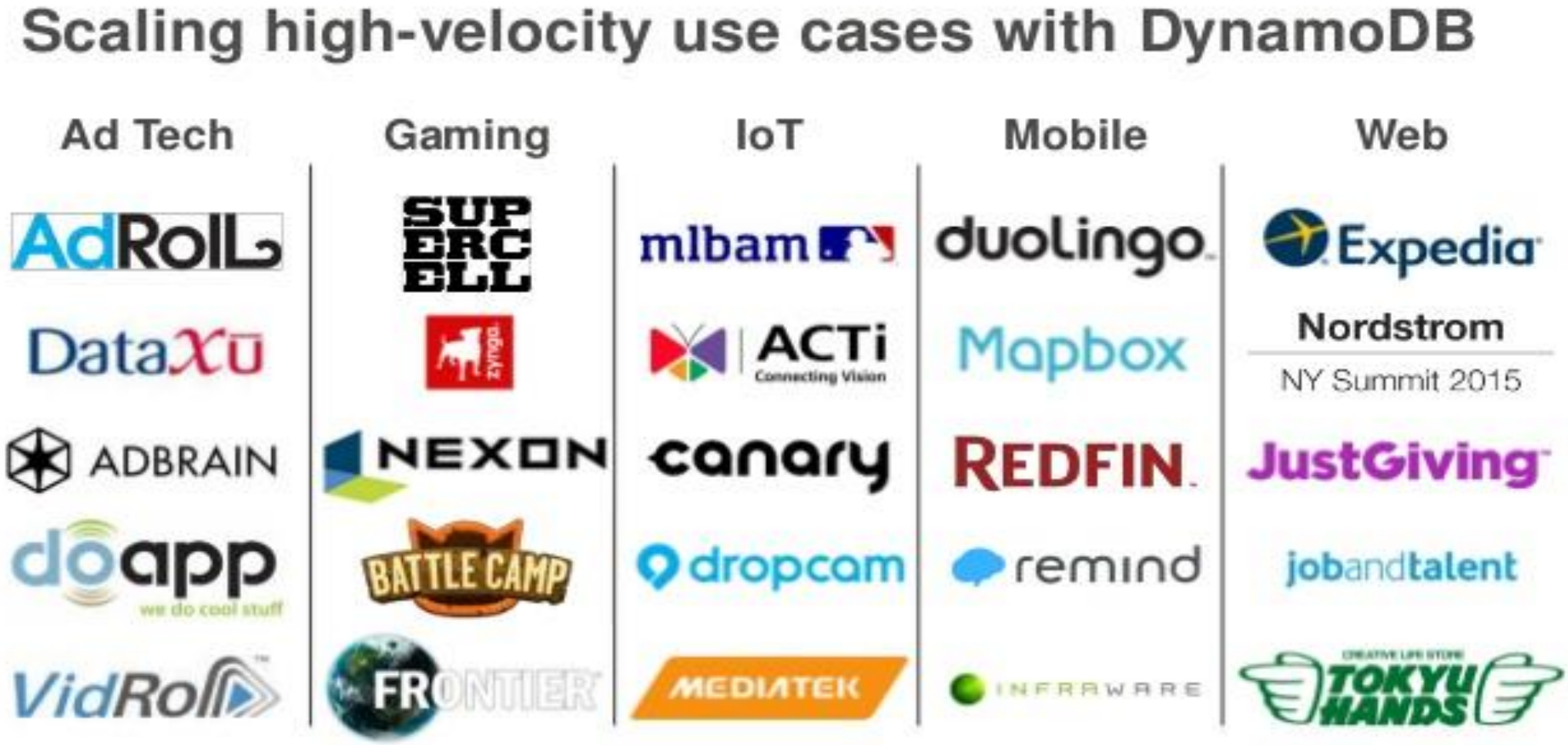
Merkle Tree
(Anti-entropy Protocol)

Performance Evaluation



- Imbalance ratio decreases with increasing load : High loads, large keys are accessed. Uniform distribution ensures load is evenly distributed.
- Low loads fewer keys are accessed, higher load imbalance.

Industrial Use Case



“ Using AWS, we can handle traffic spikes that expand up to seven times the amount of normal traffic. ”

Severin Hacker
(CTO, Duolingo)

duolingo

- Duolingo stores data about each user to be able to generate personalized lessons.
- The MySQL database couldn’t keep up with Duolingo’s rate of growth.
- By using scalable database service, data store capacity increased from 100 million to more than four billion items.
- Duolingo has the capacity to scale and support over 8 million active users.

Conclusion

- Provides incremental scalability : allowing clients to scale up and down based on their request load.
- Flexible architecture : service owners can customize their storage system to meet desired performance.
- Implements effective algorithms across clustered network for commercial platforms to perform uniform load distribution among its nodes.
- Designed to privilege write operation and handle conflicts using collapse operation using vector clocks.
- Efficient method for handling failures : Temporary failures (Hinted Handoff) and Permanent failures (Replica Synchronization).

Resources

- GG. DeCandia , D. Hastorun , M. Jampani , G. Kakulapati , A. Lakshman , A. Pilchin , S. Sivasubramanian , P.Vosshall , W. Vogels, Dynamo: amazon's highly available key-value store, Proceedings of twenty-first ACM SIGOPS symposium on Operating systems principles, October 14-17, 2007, Stevenson, Washington, USA

- Online Resources:
  - Amazon DynamoDB : https://www.youtube.com/watch?v=oz-7wJJ9HZ0
  - Consistent Hashing : https://www.youtube.com/watch?v=0X2PDMFLflY
  - Tutorials: https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html

- Industrial Use case (Duolingo) : https://aws.amazon.com/solutions/case-studies/duolingo-case-study-dynamodb/

Questions?