Scaling distributed Data Systems: A LinkedIn Case study

Sai Kiran Kanuri
Data Chef @LinkedIn
About me

- Currently at LinkedIn, messing with their data platforms for the past 20 months
- Previously bothered people at Walmart, Yahoo, Akamai & Standard Chartered Bank
- skanuri@linkedin.com
- https://www.linkedin.com/in/saikirankanur/i/
- https://www.flickr.com/saikiranrgda
Today’s agenda

<table>
<thead>
<tr>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brief Intro to Espresso</td>
</tr>
<tr>
<td>Challenges</td>
</tr>
<tr>
<td>Approaches we taken</td>
</tr>
<tr>
<td>Future Enhancements</td>
</tr>
<tr>
<td>Additional Reading</td>
</tr>
</tbody>
</table>
Espresso

- A online, distributed, fault tolerant NoSQL DB
- Multi Master cross colo support
- Bridges gap between RDBMS & k-v stores
- Hosts some of the most heavily accessed and valuable datasets at LinkedIn
- Replicates data for global availability & geo-locality.
Espresso in LinkedIn
Espresso Architecture

- Applications
- Router
- Router
- Router
- Storage-node
- Storage-node
- Data Replicator
- Cross Colo
- Kafka (Change Capture)
- Zookeeper
- Helix (Cluster Manager)
- Control Plane
- Snapshot service (Offline processing)
- Backup

Kafka

Data Replicator

Cross Colo

Kafka (Change Capture)

Zookeeper

Routing

Storage-node
Espresso Storage Node

Resource
(Database, Store, Index, Topic,..)

Partitions
p1 p2 p3 p4 p5 p6 p7 p8

Replicas

Assignments
Node-1: p1 p2 p5 p7 p6
Node-2: p3 p4 p7 p1 p8
Node-3: p5 p6 p7 p1 p3
Node-4: p7 p8 p3 p5 p4 p6
What is the biggest Challenge?
Data!
Challenges

- Interactive nature of user requests
- Low latency requirements
- Highly Volatile nature of social media requests
Challenges

Fault Tolerance

Multi-tenancy
Challenges in Fault Tolerance

- Nodes fail all the time
- Failover time
- Data replication
- Partition movement during failures
Approach to Fault Tolerance

- Minimize resource sizes
- Distribute data movement across cluster
- Minimize data movement
- Throttle data migration
- Minimize latency in control events
Approach to Fault Tolerance

- Utilize nocturnal traffic patterns for system maintenances
- Load balancing with error back-off
- Revisit read after write consistency requirements for different clients
- Enable client re-tries based on response codes
Challenges in Multi-Tenancy

- Security
- Data growth
- Hot partitions
- Service Discovery
Approach to Multi-Tenancy

- Ensure tenant isolation
- Quotas
- Transparently migrate data from one cluster to another
- Be able to re-partition existing data
- Data movement should be transparent to customer.
- Have a schema review process in place.
Future Changes

- Weighed nodes & partitions
- Dynamic data re-partitioning
- Automatic cross cluster data rebalancer
- Improved MTTR with P2P communication
Additional Reading

- R2D2 - https://github.com/linkedin/rest.li
Acknowledgements

• oakleyoriginals for funny Trash Dogs picture.

• The SCADS Director: Scaling a Distributed Storage System Under Stringent Performance Requirement paper by Beth Trushkowsky, Peter Bodík, Armando Fox, Michael J. Franklin, Michael I. Jordan, David A. Patterson

• Too Big to Fail by Kode Vicious
Questions
Thank You