Data Infrastructure @ LinkedIn

Sid Anand

QCon London 2012

@r39132

About Me

Current Life…

- LinkedIn *
  - Web / Software Engineering
    - Search, Network, and Analytics (SNA)
      - Distributed Data Systems (DDS)
        - Me

In a Previous Life…

- **Netflix**, Cloud Database Architect
- **eBay**, Web Development, Research Lab, & Search Engine ***

And Many Years Prior…

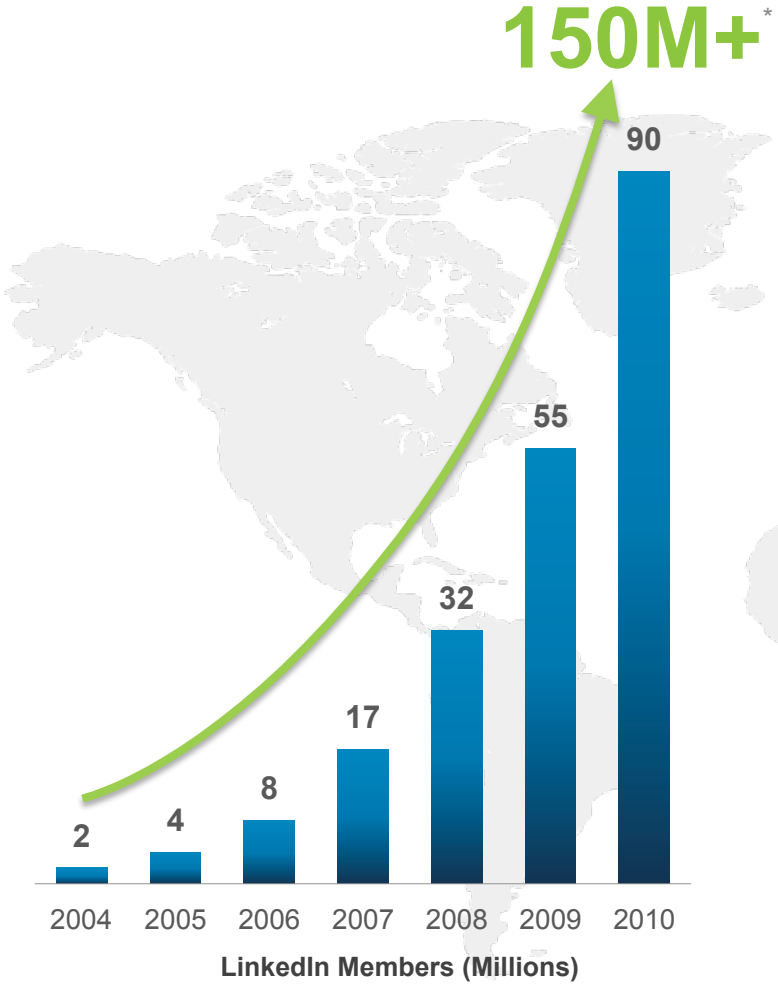
- Studying Distributed Systems at Cornell University

Our mission

Connect the world's professionals to make them more productive and successful

The world's largest professional network

Over 60% of members are now international



75%**
Fortune 100 Companies use LinkedIn to hire

>2M***
Company Pages

16
Languages

~4.2B***
Professional searches in 2011

*as of February 9, 2012
**as of September 30, 2011
***as of December 31, 2011

Other Company Facts

- Headquartered in Mountain View, Calif., with offices around the world!

- As of December 31, 2011, LinkedIn has 2,116 full-time employees located around the world.
  - Currently around 650 people work in Engineering
    - 400 in Web/Software Engineering
      - Plan to add another 200 in 2012
    - 250 in Operations

***

*as of February 9, 2012
**as of September 30, 2011
***as of December 31, 2011

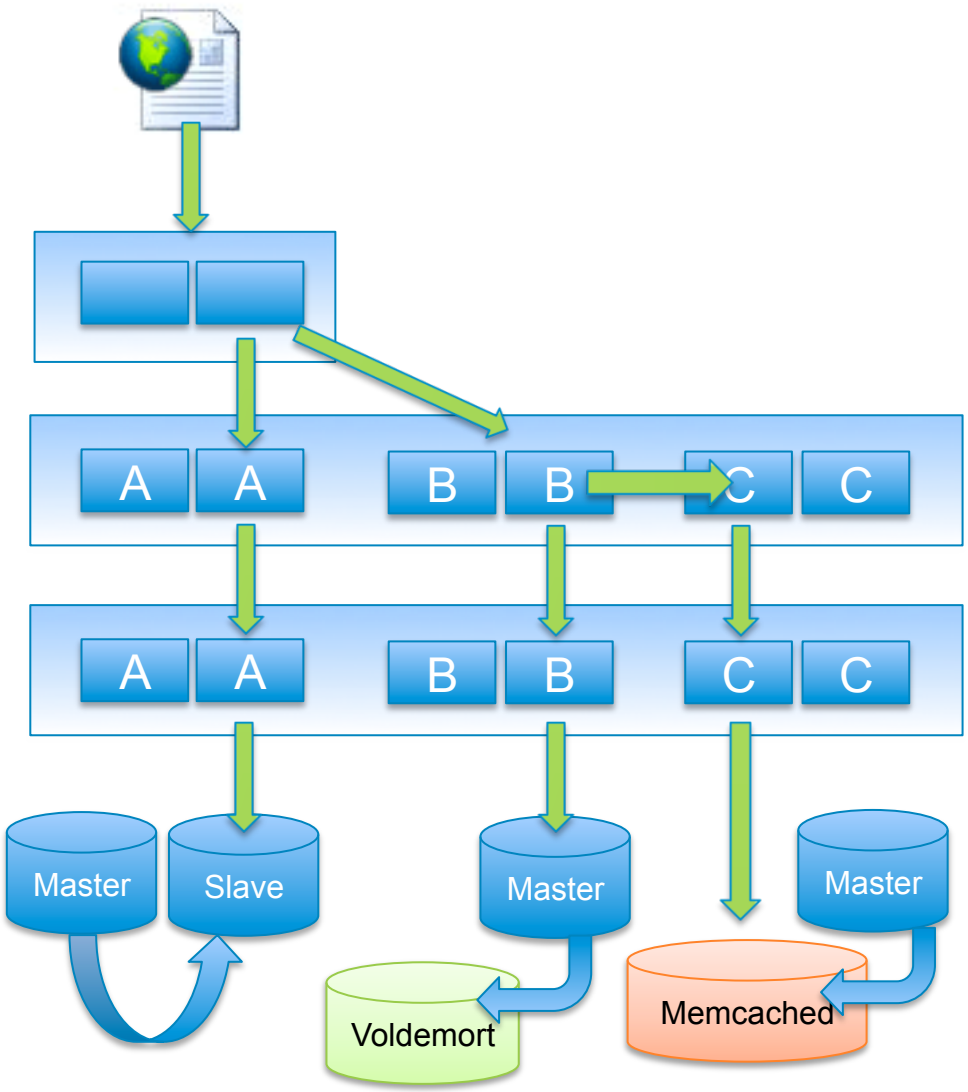
Agenda

- ✓ Company Overview
- Architecture
  - Data Infrastructure Overview
  - Technology Spotlight
    - Oracle
    - Voldemort
    - DataBus
    - Kafka
- Q & A

Overview

- Our site runs primarily on Java, with some use of Scala for specific infrastructure
- What runs on Scala?
  - Network Graph Service
  - Kafka
- Most of our services run on Apache + Jetty

LinkedIn : Architecture



→ A web page requests information A and B

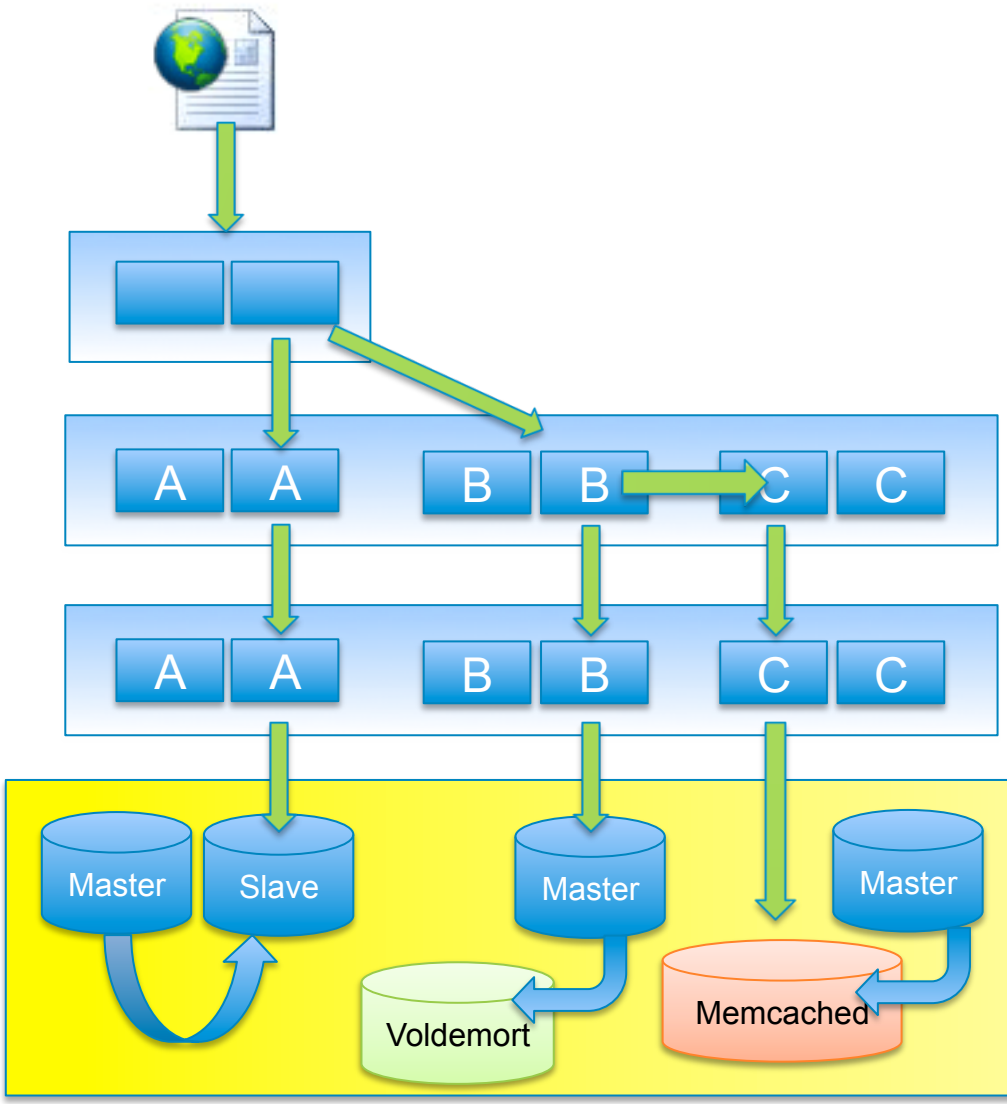
Presentation Tier → A thin layer focused on building the UI. It assembles the page by making parallel requests to BST services

Business Service Tier → Encapsulates business logic. Can call other BST clusters and its own DST cluster.

Data Service Tier → Encapsulates DAL logic and concerned with one Oracle Schema.

Data Infrastructure → Concerned with the persistent storage of and easy access to data

LinkedIn : Architecture



→ A web page requests information A and B

Presentation Tier → A thin layer focused on building the UI. It assembles the page by making parallel requests to BST services

Business Service Tier → Encapsulates business logic. Can call other BST clusters and its own DST cluster.

Data Service Tier → Encapsulates DAL logic and concerned with one Oracle Schema.

Data Infrastructure → Concerned with the persistent storage of and easy access to data

LinkedIn : Data Infrastructure Technologies

- Database Technologies
  - Oracle
  - Voldemort *
  - Espresso

- Data Replication Technologies
  - Kafka
  - DataBus

- Search Technologies
  - Zoie – real-time search and indexing with Lucene
  - Bobo – faceted search library for Lucene ***
  - SenseiDB – fast, real-time, faceted, KV and full-text Search Engine and more

LinkedIn : Data Infrastructure Technologies

This talk will focus on a few of the key technologies below!

- Database Technologies *
  - Oracle
  - Voldemort
  - Espresso – A new K-V store under development

- Data Replication Technologies
  - Kafka
  - DataBus

- Search Technologies ***
  - Zoie – real-time search and indexing with Lucene
  - Bobo – faceted search library for Lucene
  - SenseiDB – fast, real-time, faceted, KV and full-text Search Engine and more

LinkedIn Data Infrastructure Technologies

Oracle: Source of Truth for User-Provided Data

Oracle : Overview

Oracle

- All user-provided data is stored in Oracle – our current source of truth
- About 50 Schemas running on tens of physical instances
- With our user base and traffic growing at an accelerating pace, so how do we scale Oracle for user-provided data?

Scaling Reads

- Oracle Slaves (**c.f. DSC**)
- Memcached
- Voldemort – for key-value lookups

***

Scaling Writes

- Move to more expensive hardware **or** replace Oracle with something else

Scaling Oracle Reads using DSC

- DSC uses a token (e.g. cookie) to ensure that a reader always sees his or her own writes immediately
  - If I update my own status, it is okay if you don't see the change for a few minutes, but I have to see it immediately

Oracle : Overview – How DSC Works?

- When a user writes data to the master, the DSC token (for that data domain) is updated with a timestamp
- When the user reads data, we first attempt to read from a replica (a.k.a. slave) database
- If the data in the slave is older than the data in the DSC token, we read from the Master instead

LinkedIn Data Infrastructure Technologies

Voldemort: Highly-Available Distributed Data Store

Voldemort : Overview

- A distributed, persistent key-value store influenced by the AWS Dynamo paper

- Key Features of Dynamo
  - Highly Scalable, Available, and Performant
  - Achieves this via Tunable Consistency
    - For higher consistency, the user accepts lower availability, scalability, and performance, and vice-versa

  - Provides several self-healing mechanisms when data does become inconsistent
    - Read Repair
      - Repairs value for a key when the key is looked up/read
    - Hinted Handoff
      - Buffers value for a key that wasn't successfully written, then writes it later
    - Anti-Entropy Repair
      - Scans the entire data set on a node and fixes it

  - Provides means to detect node failure and a means to recover from node failure
    - Failure Detection
    - Bootstrapping New Nodes

Voldemort : Overview

API

VectorClock<V> **get** (K key)

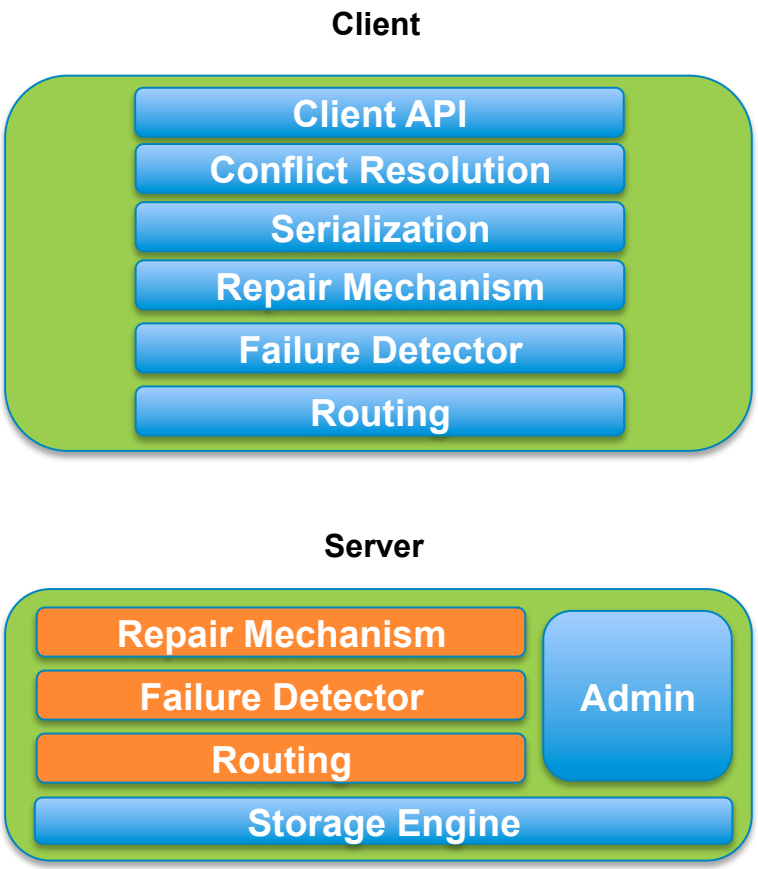
put (K key, VectorClock<V> value)

applyUpdate(UpdateAction action, int retries)

Voldemort-specific Features

- Implements a layered, pluggable architecture
- Each layer implements a common interface (c.f. API). This allows us to replace or remove implementations at any layer
  - Pluggable data storage layer
    - BDB JE, Custom RO storage, etc…
  - Pluggable routing supports
    - Single or Multi-datacenter routing

Layered, Pluggable Architecture

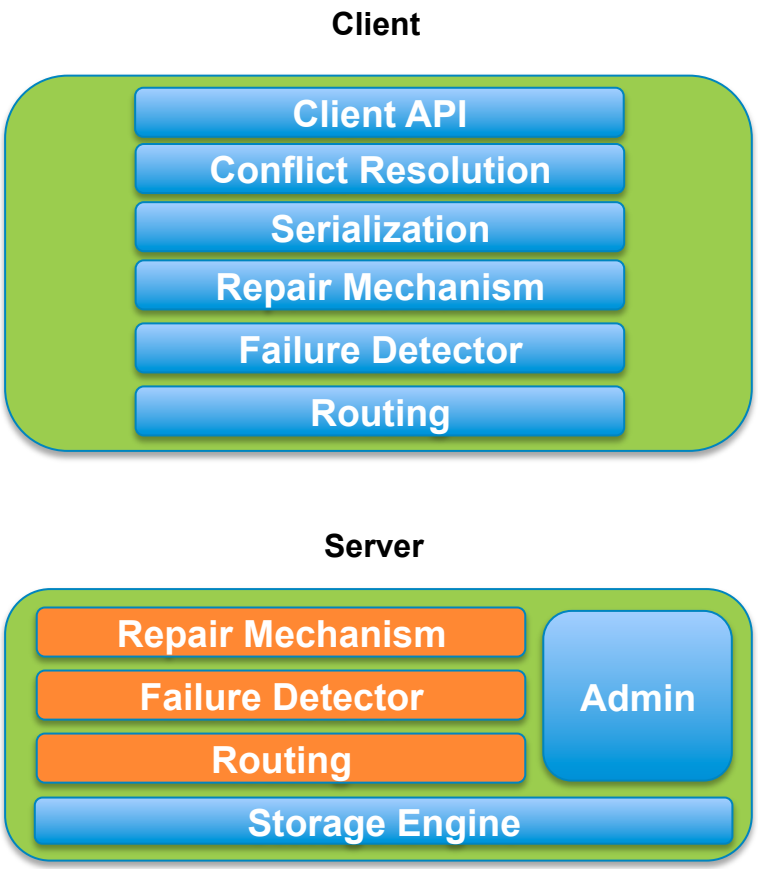


Voldemort : Overview

Voldemort-specific Features

- Supports Fat client or Fat Server
  - **Repair Mechanism** + **Failure Detector** + **Routing** can run on server or client
- LinkedIn currently runs Fat Client, but we would like to move this to a Fat Server Model

Layered, Pluggable Architecture



Where Does LinkedIn use Voldemort?

Voldemort : Usage Patterns @ LinkedIn

2 Usage-Patterns

- Read-Write Store
  - Uses BDB JE for the storage engine
  - 50% of Voldemort Stores (aka Tables) are RW
- Read-only Store
  - Uses a custom Read-only format
  - 50% of Voldemort Stores (aka Tables) are RO
- Let's look at the RO Store

Voldemort : RO Store Usage at LinkedIn

People You May Know

People You May Know

Roshan Sumbaly, Senior Software Engineer at LinkedIn
Connect

Alex Feinberg, Senior Software Engineer at LinkedIn
Connect

Jay Kreps, Principal Staff Engineer at LinkedIn
Connect

See more »

Viewers of this profile also viewed

Viewers of this profile also viewed...

Sam Shah
Principal Engineer at LinkedIn

Igor Perisic
Director of Engineering; Search,...

Anmol Bhasin
Recommendations, A/B Testing and...

Jun Rao
Principle Software Engineer at LinkedIn

Related Searches

Related searches for **hadoop**

mapreduce
big data
machine learning
data mining
java
hbase
lucene
data warehouse

Events you may be interested in

Events you may be interested in — Browse Internet events »

Improving Hadoop Performance by (up to) 1000x - A Linkedin Te…
December 13, 2011 – LinkedIn headquarters - TALK OPEN TO PUBLIC, Mount…
and 251 other people are attending.

2612 Introduction to Machine Learning and Data Mining
January 31, 2012 – University of California - Santa Cruz Extension in Santa Clar…
and 9 other people are attending.

Ninth Software Craftsmanship Meeting
December 19, 2011 – SAP Labs, HaTidhar 15 Ra'anana, 43665, Israel
are attending.

3rd Italian Information Retrieval Workshop (IIR 2012)
January 26-27, 2012 – Dipartimento di Informatica (DIB), Università di Bari "Ald…
and 4 other people are attending.

Clojure/West 2012
March 16-17, 2012 – San Jose Marriott
and 10 other people are attending.

LinkedIn Skills

Skills & Expertise Beta > Hadoop

Search Skills & Expertise

Related Skills: HBase, MapReduce, Nutch, Solr, Lucene, AWS, EC2, Collaborative Filtering, Amazon Web Services, RDFS, Weka, Recommender Systems, Clojure

Hadoop ▲33% y/y

Primary Industry: Internet

Apache Hadoop is a Java software framework that supports data-intensive distributed applications under a free license. It enables applications to work with thousands of nodes and petabytes of data. Hadoop was inspired by Google's MapReduce and Google File System (GFS) papers. Hadoop is a top-level Apache project, being built and used by a community of…

More on 'Hadoop' at Wikipedia »

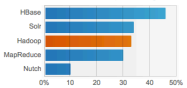
✓ Listed on your profile — Edit Your Skills

Hadoop Professionals

Arun C Murthy (2nd)
Founder and Architect at Hortonworks Inc., VP Apache Hadoop at ASF
I am a Founder and Architect at Hortonworks Inc. Hortonworks was formed by the key architects and core Hadoop…

Share | Tweet

Relative Growth | Size | Age



Related Companies

The Apache Software Foundation
Computer Software, United States
Follow

Cloudera
Computer Software, San Francisco Bay Area
Stop following

Jobs you may be interested in

Jobs you may be interested in Beta — Email Alerts | See More »

Senior Software Engineer – Applications
Modicom - San Francisco Bay Area

Senior Software Engineer, C/C++
StumbleUpon - San Francisco, Ca

Sr. R&D Java Software Engineer - Rare and unique start-up
Medallia, Inc. - Palo Alto, CA

Senior Software Engineer
CyberCoders - San Jose ,CA

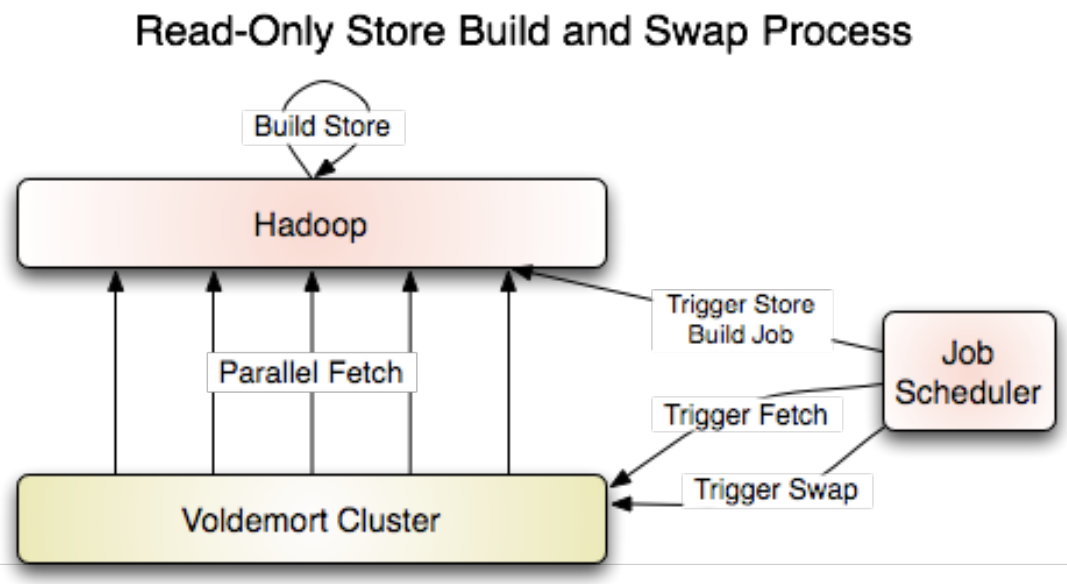
Senior Software Engineer - Qualcomm Platform
Pelican Imaging Corporation - San Francisco Bay Area

RO Store Usage Pattern

1. Use Hadoop to build a model

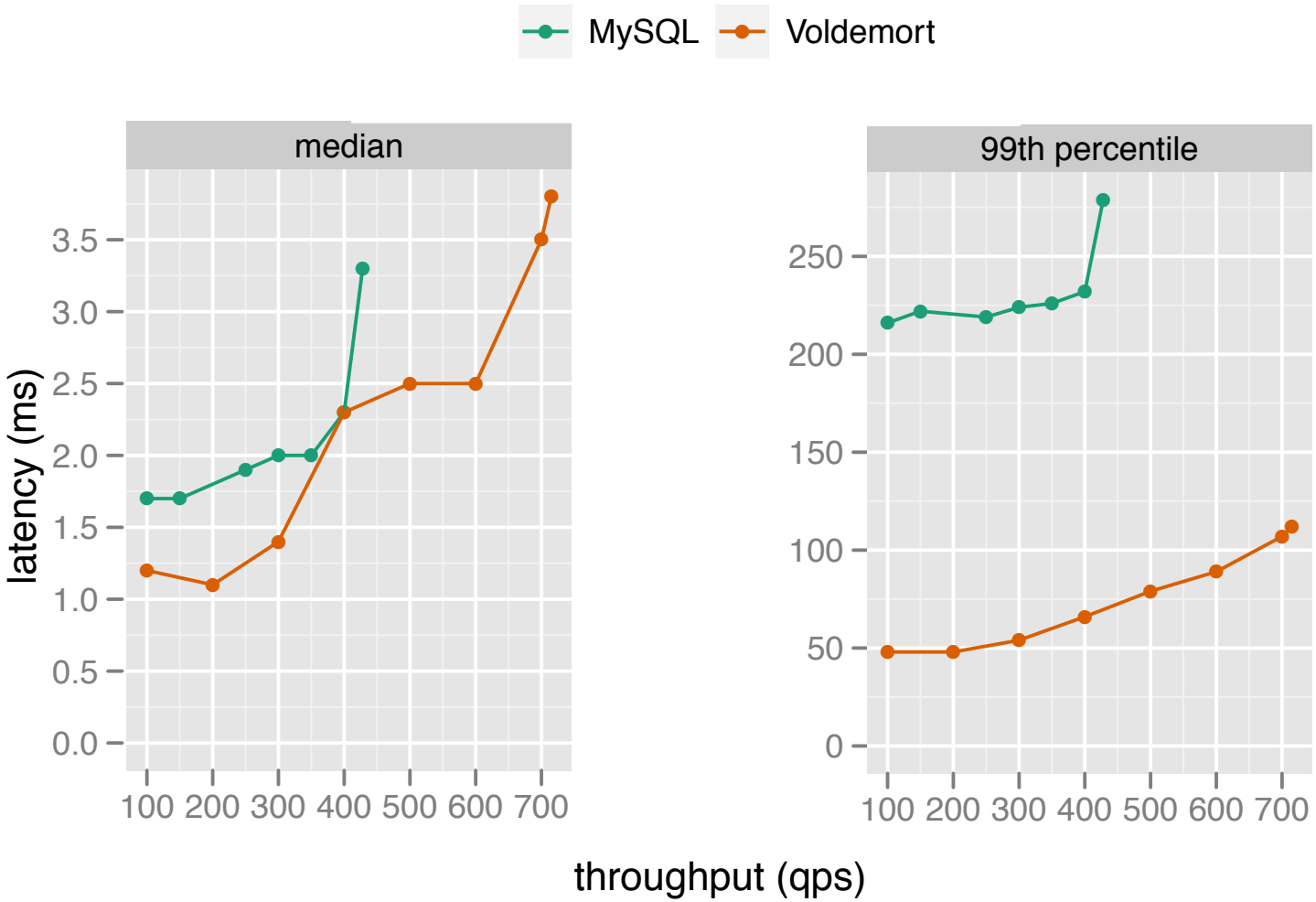
2. Voldemort loads the output of Hadoop

3. Voldemort serves fast key-value look-ups on the site
   - e.g. For key="Sid Anand", get all the people that "Sid Anand" may know!
   - e.g. For key="Sid Anand", get all the jobs that "Sid Anand" may be interested in!



How Do The Voldemort RO Stores Perform?

Voldemort : RO Store Performance : TP vs. Latency



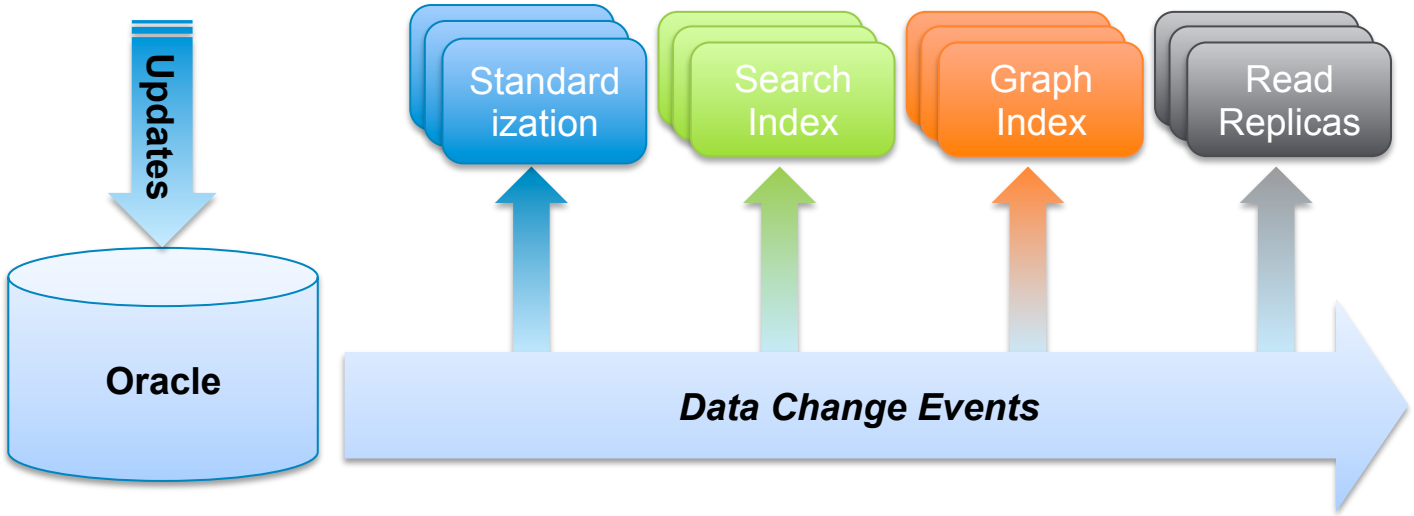
100 GB data, 24 GB RAM

LinkedIn Data Infrastructure Solutions

Databus : Timeline-Consistent Change Data Capture

Where Does LinkedIn use DataBus?

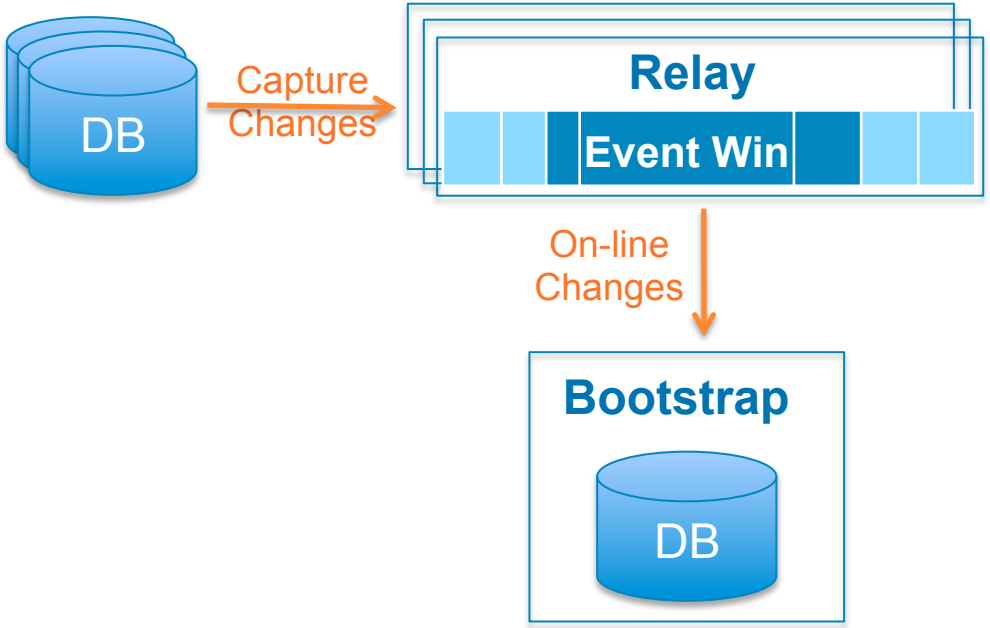
DataBus : Use-Cases @ LinkedIn



A user updates his profile with skills and position history. He also accepts a connection

- The write is made to an Oracle Master and DataBus replicates:
- the profile change to the Standardization service
  - E.G. the many forms of IBM are canonicalized for search-friendliness and recommendation-friendliness
- the profile change to the Search Index service
  - Recruiters can find you immediately by new keywords
- the connection change to the Graph Index service
  - The user can now start receiving feed updates from his new connections immediately

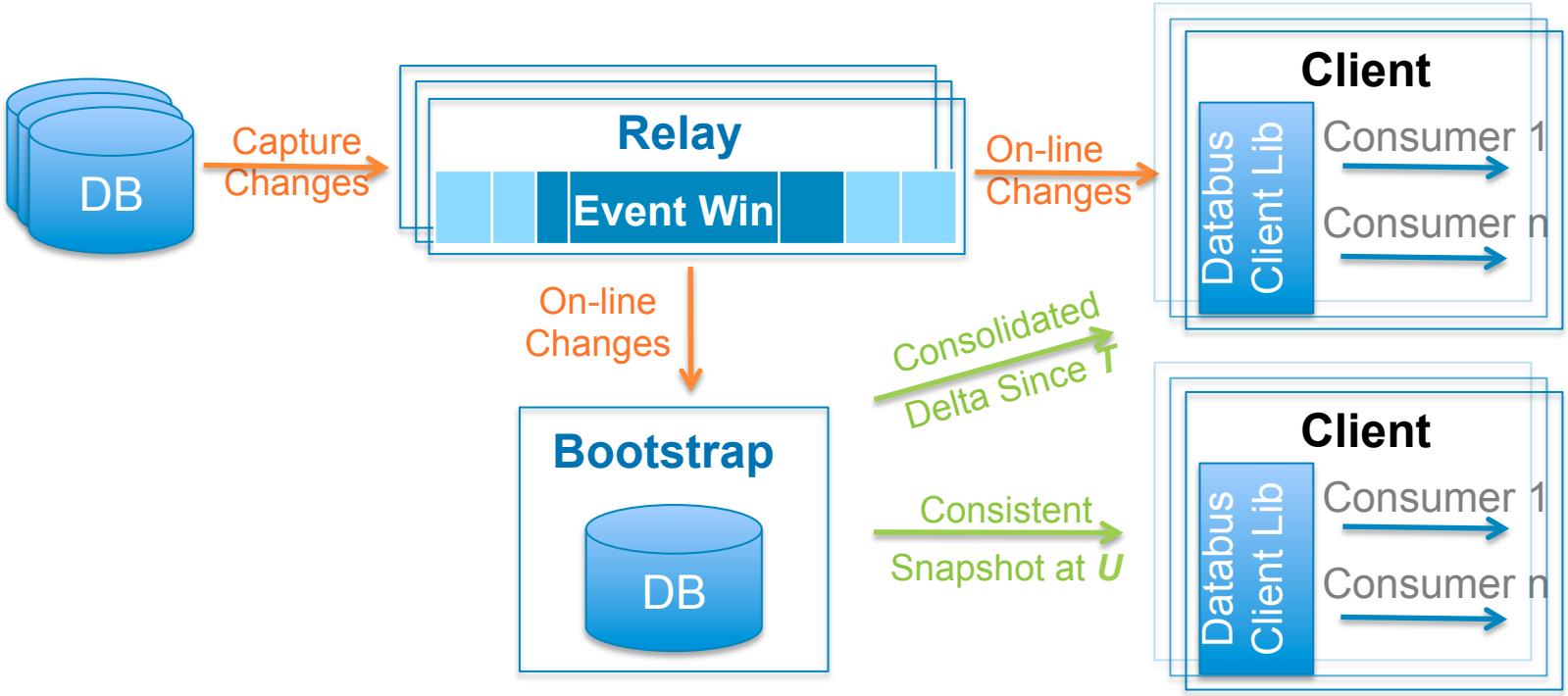
DataBus : Architecture



DataBus consists of 2 services

- **Relay Services**
  - Sharded
  - Maintain an in-memory buffer per shard
  - Each shard polls Oracle and then deserializes transactions into Avro
- **Bootstrap Service**
  - Picks up online changes as they appear in the Relay
  - Supports 2 types of operations from clients
    - ➢ If a client falls behind and needs records older than what the relay has, Bootstrap can send consolidated deltas!
    - ➢ If a new client comes on line, Bootstrap can send a consistent snapshot

DataBus : Architecture

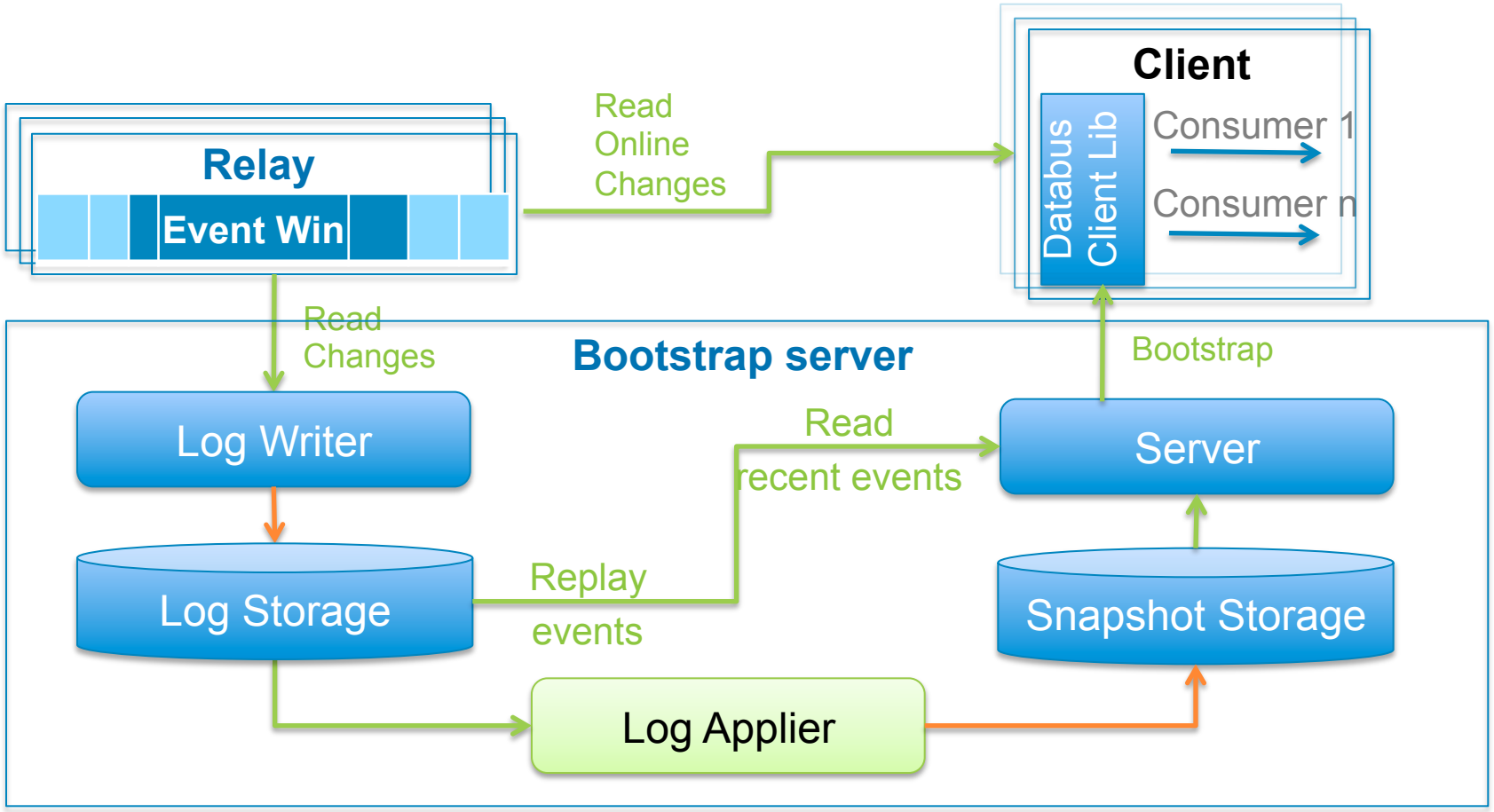


Guarantees

- Transactional semantics
- In-commit-order Delivery
- At-least-once delivery
- Durability (by data source)
- High-availability and reliability
- Low latency

DataBus : Architecture - Bootstrap

- Generate consistent snapshots and consolidated deltas during continuous updates with long-running queries



LinkedIn Data Infrastructure Solutions

Kafka: High-Volume Low-Latency Messaging System

Kafka : Usage at LinkedIn

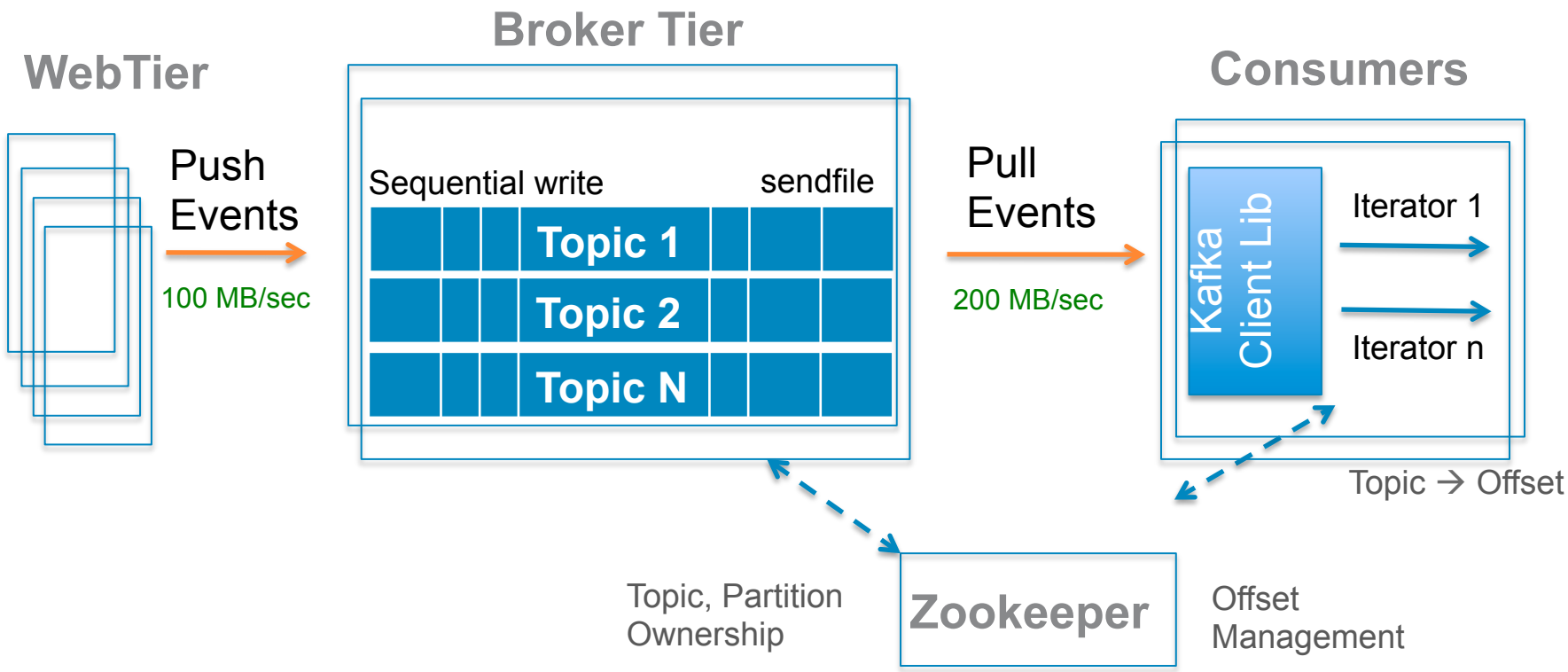
Where as DataBus is used for Database change capture and replication, Kafka is used for application-level data streams

Examples:

- End-user Action Tracking (a.k.a. Web Tracking) of
  - Emails opened
  - Pages seen
  - Links followed
  - Executing Searches

- Operational Metrics
  - Network & System metrics such as
    - TCP metrics (connection resets, message resends, etc…)
    - System metrics (iops, CPU, load average, etc…)

Kafka : Overview



Features

- Pub/Sub
- Batch Send/Receive
- System Decoupling

Guarantees

- At least once delivery
- Very high throughput
- Low latency
- Durability
- Horizontally Scalable

Scale

- Billions of Events
- TBs per day
- Inter-colo: few seconds
- Typical retention: weeks

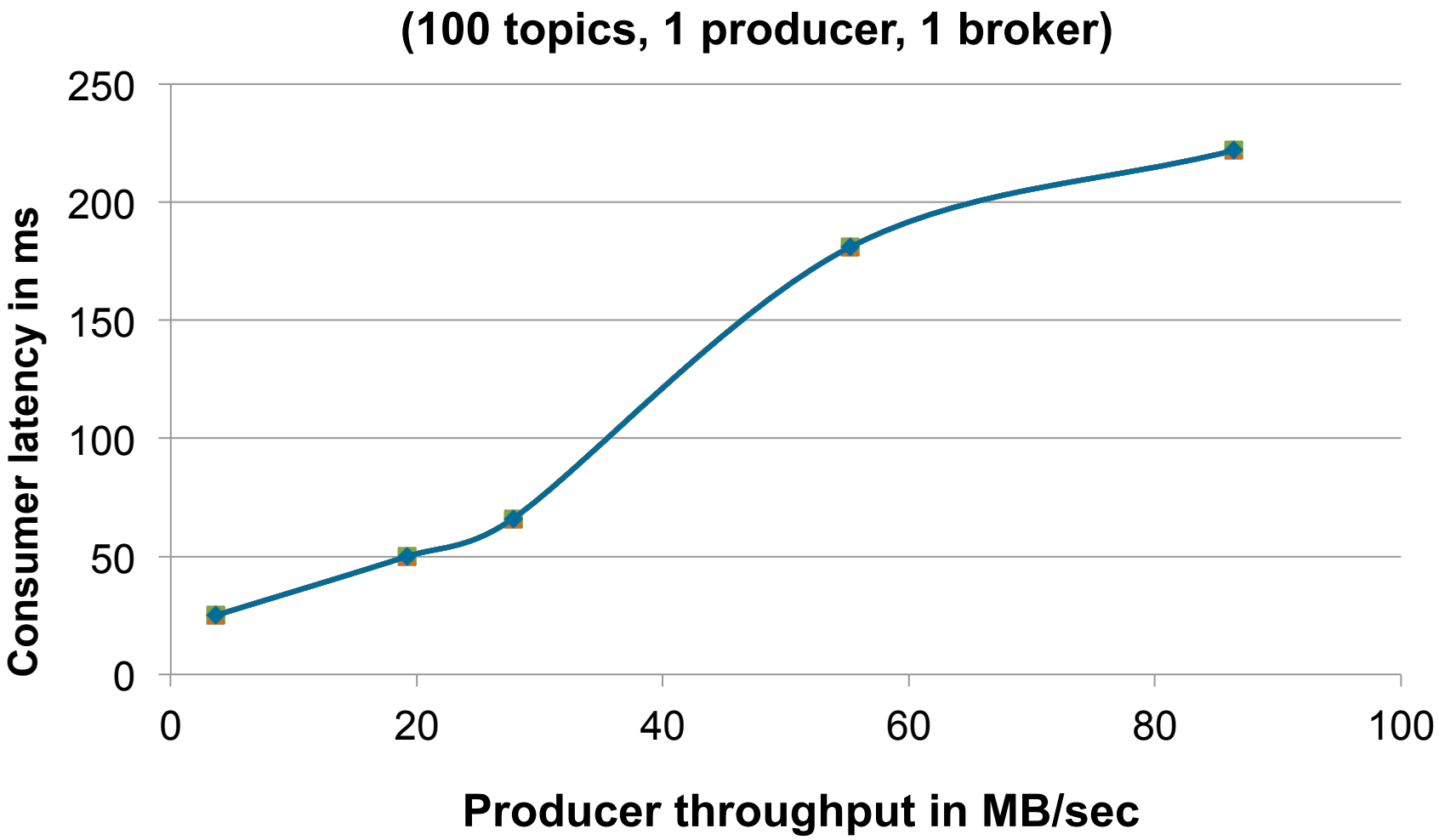
Kafka : Overview

Key Design Choices

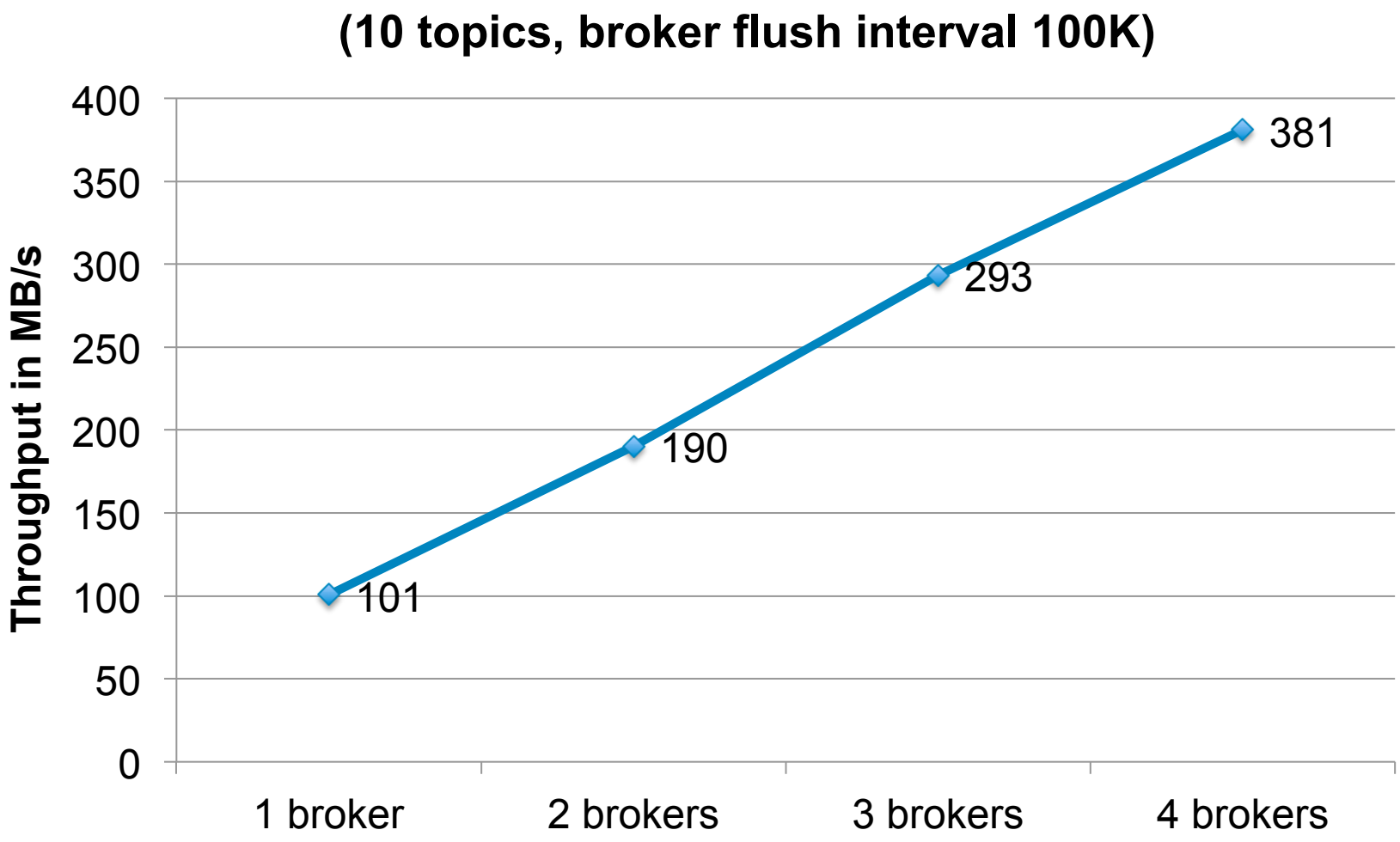
- When reading from a file and sending to network socket, we typically incur **4 buffer copies** and **2 OS system calls**
  - Kafka leverages a **sendFile** API to eliminate 2 of the buffer copies and 1 of the system calls
- No double-buffering of messages - we rely on the OS page cache and do not store a copy of the message in the JVM
  - Less pressure on memory and GC
  - If the Kafka process is restarted on a machine, recently accessed messages are still in the page cache, so we get the benefit of a warm start
- Kafka doesn't keep track of which messages have yet to be consumed -- i.e. no book keeping overhead
  - Instead, messages have time-based SLA expiration -- after 7 days, messages are deleted

How Does Kafka Perform?

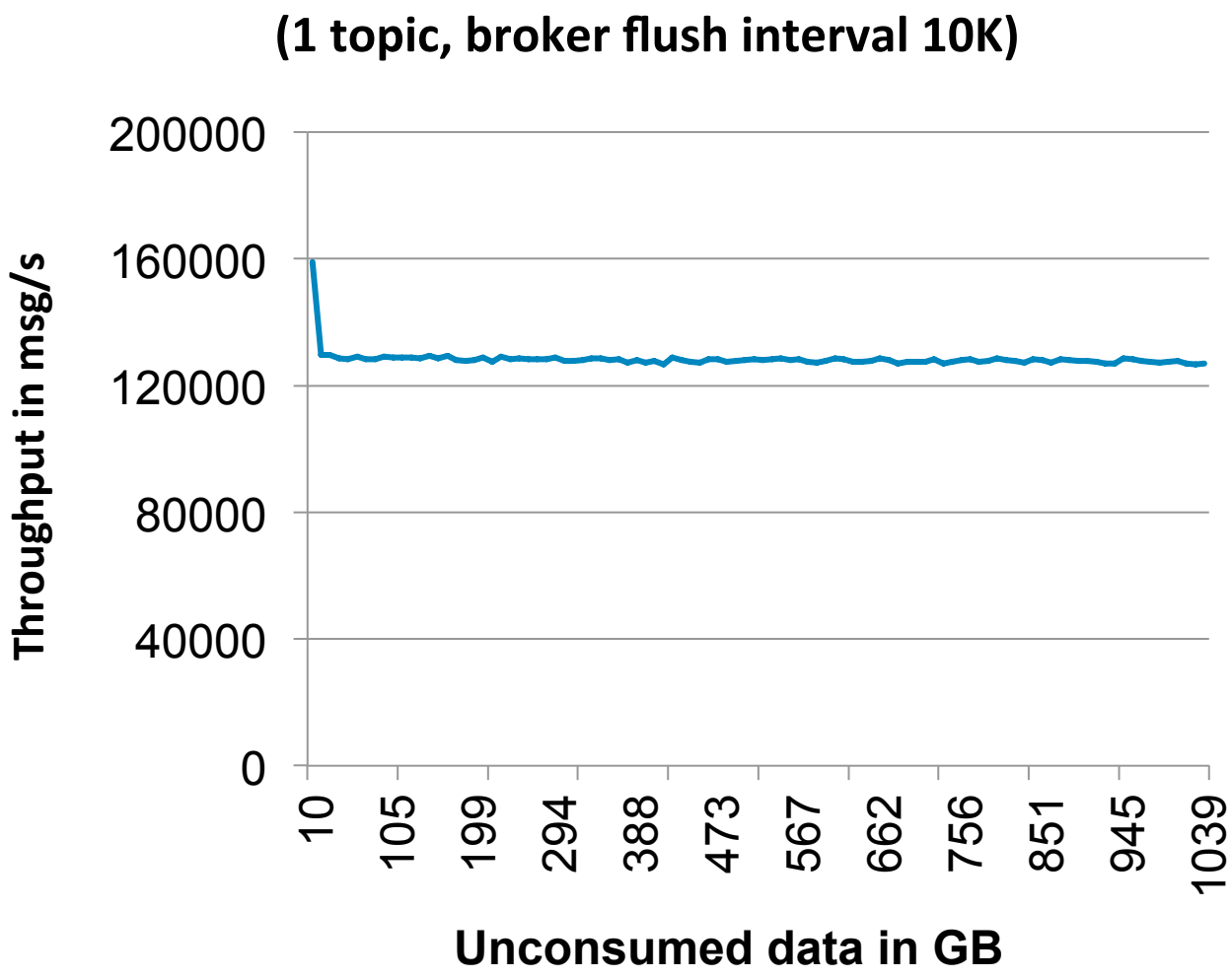
Kafka : Performance : Throughput vs. Latency



Kafka : Performance : Linear Incremental Scalability



Kafka : Performance : Resilience as Messages Pile Up



Acknowledgments

Presentation & Content

- Chavdar Botev (**DataBus**) @cbotev
- Roshan Sumbaly (**Voldemort**) @rsumbaly
- Neha Narkhede (**Kafka**) @nehanarkhede

Development Team

Aditya Auradkar, Chavdar Botev, Shirshanka Das, Dave DeMaagd, Alex Feinberg, Phanindra Ganti, Lei Gao, Bhaskar Ghosh, Kishore Gopalakrishna, Brendan Harris, Joel Koshy, Kevin Krawez, Jay Kreps, Shi Lu, Sunil Nagaraj, Neha Narkhede, Sasha Pachev, Igor Perisic, Lin Qiao, Tom Quiggle, Jun Rao, Bob Schulman, Abraham Sebastian, Oliver Seeliger, Adam Silberstein, Boris Skolnick, Chinmay Soman, Roshan Sumbaly, Kapil Surlaker, Sajid Topiwala, Balaji Varadarajan, Jemiah Westerman, Zach White, David Zhang, and Jason Zhang

y Questions?