Meeting of the LF AI & Data Technical Advisory Council (TAC)

May 4, 2023

LF AI & DATA

Antitrust Policy

- Linux Foundation meetings involve participation by industry competitors, and it is the intention of the Linux Foundation to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of, and not participate in, any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

- Examples of types of actions that are prohibited at Linux Foundation meetings and in connection with Linux Foundation activities are described in the Linux Foundation Antitrust Policy available at http://www.linuxfoundation.org/antitrust-policy. If you have questions about these matters, please contact your company counsel, or if you are a member of the Linux Foundation, feel free to contact Andrew Updegrove of the firm of Gesmer Undergone LLP, which provides legal counsel to the Linux Foundation.

Recording of Calls

Reminder:

TAC calls are recorded and available for viewing on the TAC Wiki

Reminder: LF AI & Data Useful Links

- Web site: lfaidata.foundation
- Wiki: wiki.lfaidata.foundation
- GitHub: github.com/lfaidata
- Landscape: https://landscape.lfaidata.foundation or https://l.lfaidata.foundation
- Mail Lists: https://lists.lfaidata.foundation
- Slack: https://slack.lfaidata.foundation
- Youtube: https://www.youtube.com/channel/UCfasaeqXJBCAJMNO9HcHfbA
- LF AI Logos: https://github.com/lfaidata/artwork/tree/master/lfaidata
- LF AI Presentation Template: https://drive.google.com/file/d/1eiDNJvXCqSZHT4Zk_-czASlz2GTBRZk2/view?usp=sharing

- Events Page on LF AI Website: https://lfaidata.foundation/events/
- Events Calendar on LF AI Wiki (subscribe available): https://wiki.lfaidata.foundation/pages/viewpage.action?pageId=12091544
- Event Wiki Pages: https://wiki.lfaidata.foundation/display/DL/LF+AI+Data+Foundation+Events

Agenda

- Roll Call (1 mins)
- Approval of Minutes from previous meeting (2 mins)
- Update on the project lifecycle doc (2 mins)
- Amundsen Project review (10 minutes)
- LakeSoul new project (Amir Banifatemi, 40 minutes)
- Open Discussion

TAC Voting Members - Please note

Please ensure that you do the following to facilitate smooth procedural quorum and voting processes:

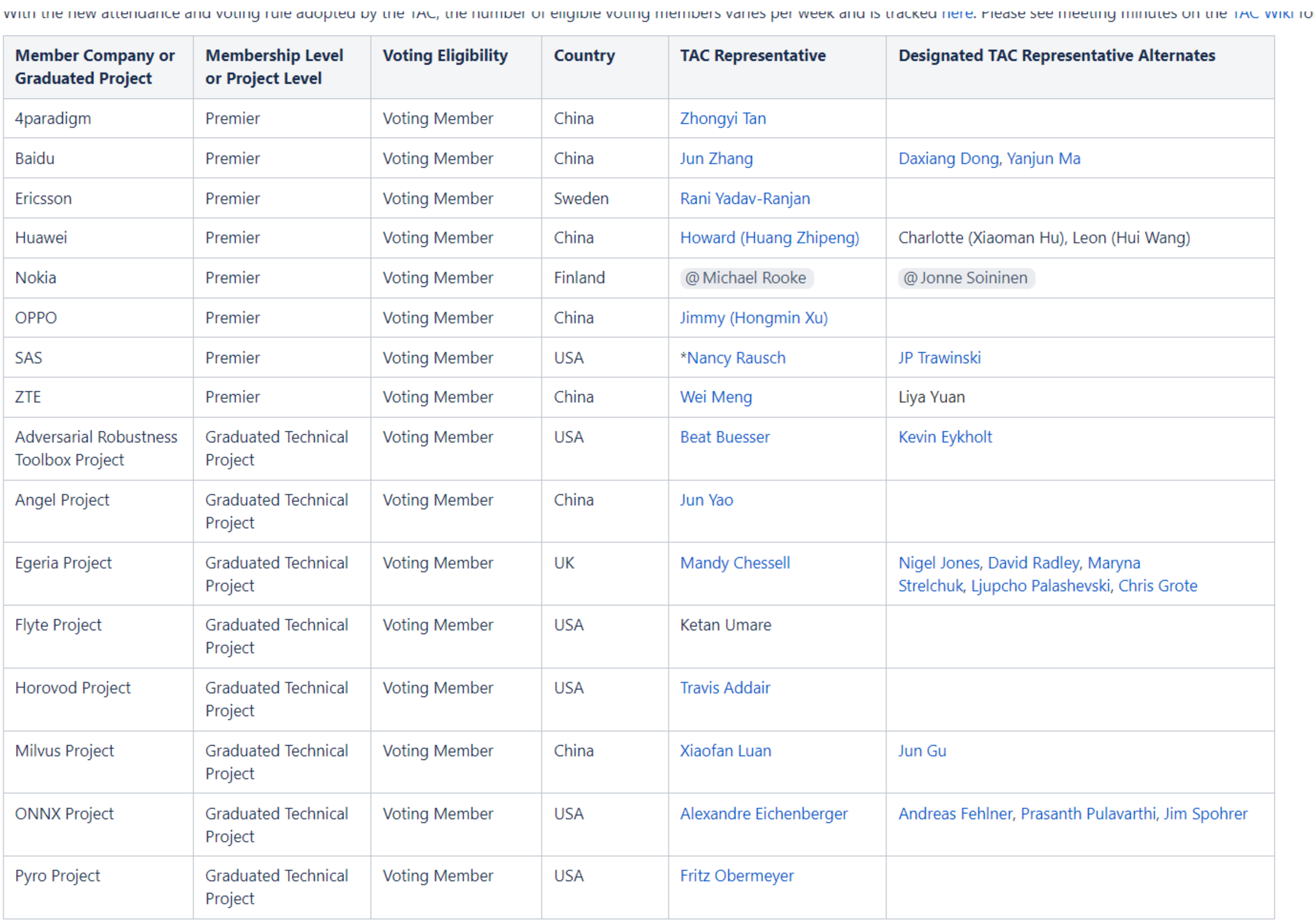
- Change your Zoom display name to include your First/Last Name, Company/Project Represented
  - example: Nancy Rausch, SAS
- State your First/Last Name and Company/Project when submitting a motion
  - example: First motion, Nancy Rausch/SAS

TAC Voting Members - Please note

- TAC members must attend consistently to maintain their voting status
- After 2 absences voting members will lose voting privileges
- Voting privileges will only be reinstated after attending 2 meetings in a row

TAC Voting Members

Note: we still need a few designated backups specified on wiki



With the new attendance and voting rule adopted by the TAC, the number of eligible voting members varies per week and is tracked here. Please see meeting minutes on the TAC wiki fo

Member Company or Graduated Project	Membership Level or Project Level	Voting Eligibility	Country	TAC Representative	Designated TAC Representative Alternates
4paradigm	Premier	Voting Member	China	Zhongyi Tan	
Baidu	Premier	Voting Member	China	Jun Zhang	Daxiang Dong, Yanjun Ma
Ericsson	Premier	Voting Member	Sweden	Rani Yadav-Ranjan	
Huawei	Premier	Voting Member	China	Howard (Huang Zhipeng)	Charlotte (Xiaoman Hu), Leon (Hui Wang)
Nokia	Premier	Voting Member	Finland	@Michael Rooke	@Jonne Soininen
OPPO	Premier	Voting Member	China	Jimmy (Hongmin Xu)	
SAS	Premier	Voting Member	USA	*Nancy Rausch	JP Trawinski
ZTE	Premier	Voting Member	China	Wei Meng	Liya Yuan
Adversarial Robustness Toolbox Project	Graduated Technical Project	Voting Member	USA	Beat Buesser	Kevin Eykholt
Angel Project	Graduated Technical Project	Voting Member	China	Jun Yao	
Egeria Project	Graduated Technical Project	Voting Member	UK	Mandy Chessell	Nigel Jones, David Radley, Maryna Strelchuk, Ljupcho Palashevski, Chris Grote
Flyte Project	Graduated Technical Project	Voting Member	USA	Ketan Umare	
Horovod Project	Graduated Technical Project	Voting Member	USA	Travis Addair	
Milvus Project	Graduated Technical Project	Voting Member	China	Xiaofan Luan	Jun Gu
ONNX Project	Graduated Technical Project	Voting Member	USA	Alexandre Eichenberger	Andreas Fehlner, Prasanth Pulavarthi, Jim Spohrer
Pyro Project	Graduated Technical Project	Voting Member	USA	Fritz Obermeyer	

Minutes approval

LF AI & DATA

Approval of April 6, 2023 Minutes

Draft minutes from the April 6 TAC call were previously distributed to the TAC members via the mailing list

Proposed Resolution:

- That the minutes of the April 6 meeting of the Technical Advisory Council of the LF AI & Data Foundation are hereby approved.

Reminder – Please review the Project Lifecycle Doc

- We have received many updates and all changes have been incorporated
- Please review
- We will have a vote on May 18 to approve the doc during the next TAC call

- Link: https://docs.google.com/document/d/1VFoMcXlbOKa4cqTLkaWX-8_7-uh11hZT3b6p0wvjUHM/edit#heading=h.gjdgxs

Annual Review for Amundsen

May 4, 2023

LF AI & DATA

Amundsen

Amundsen

Brief Description:

Amundsen is a data discovery and metadata engine for improving the productivity of data analysts, data scientists and engineers when interacting with data. It does that today by indexing data resources (tables, dashboards, streams, etc.) and powering a page-rank style search based on usage patterns (e.g. highly queried tables show up earlier than less queried tables). Think of it as Google search for data.

Contributed by:

Lyft in July 2020 as an Incubation Project

Key Links:

Github: https://github.com/amundsen-io

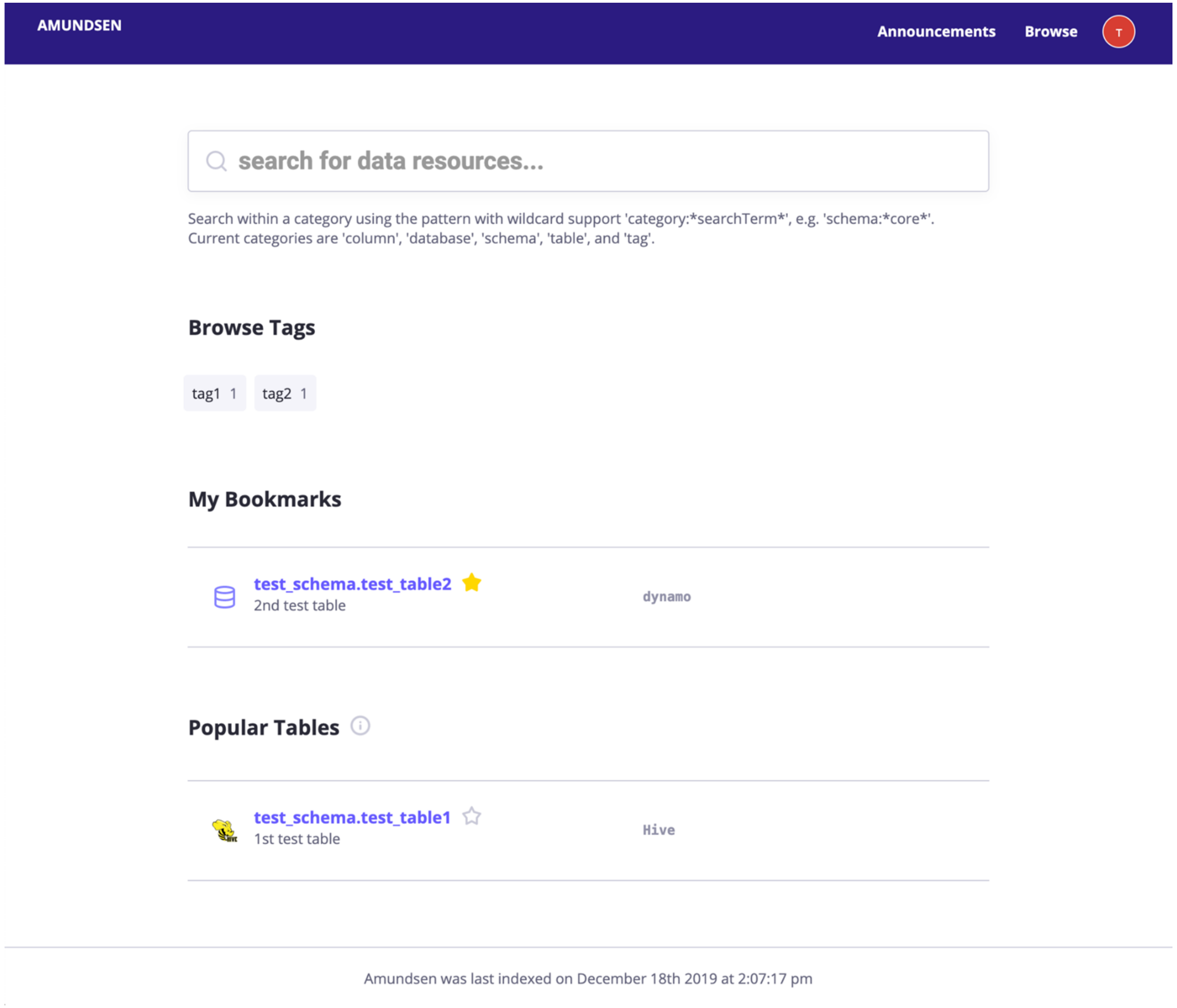
Website: https://www.amundsen.io/

Artwork:

https://github.com/lfai/artwork/tree/master/projects/amundsen

Mailing lists:

- amundsen-announce
- amundsen-technical-discuss
- amundsen-tsc



AMUNDSEN

Announcements

Browse

T

search for data resources...

Search within a category using the pattern with wildcard support 'category:*searchTerm*', e.g. 'schema:*core*'. Current categories are 'column', 'database', 'schema', 'table', and 'tag'.

Browse Tags

tag1 1 tag2 1

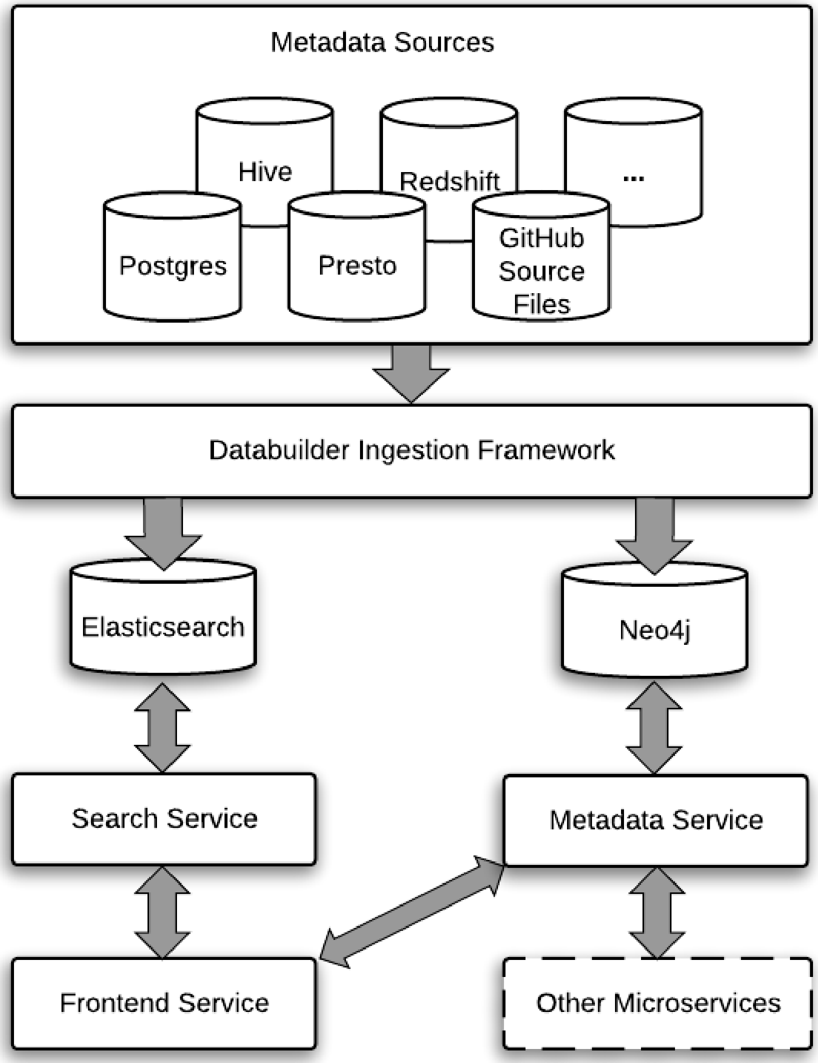
My Bookmarks

test_schema.test_table2 ★
2nd test table
dynamo

Popular Tables ⓘ

test_schema.test_table1 ☆
1st test table
Hive

Amundsen was last indexed on December 18th 2019 at 2:07:17 pm



Organizations contributing

lyft

Delivery Hero

DevotedHealth

ING

databricks

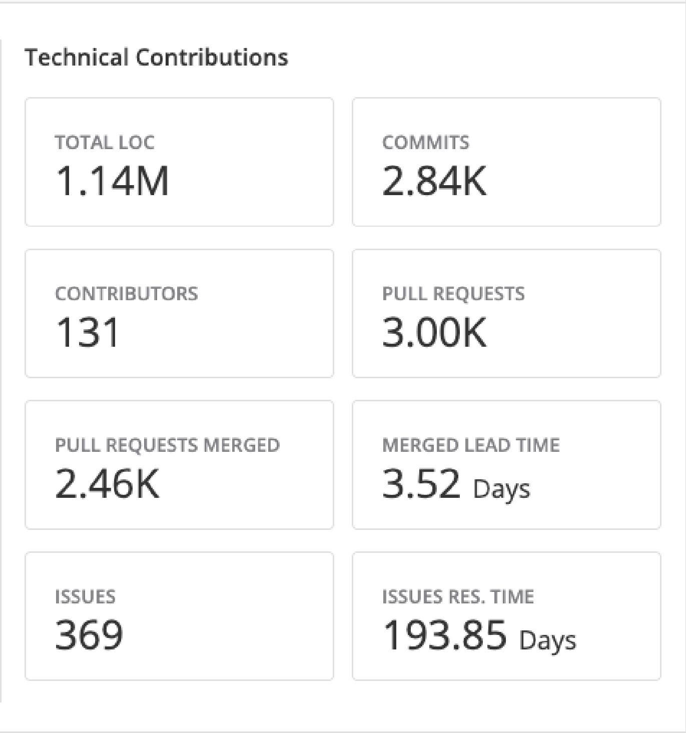
intel®

SAMSUNG

PLAID

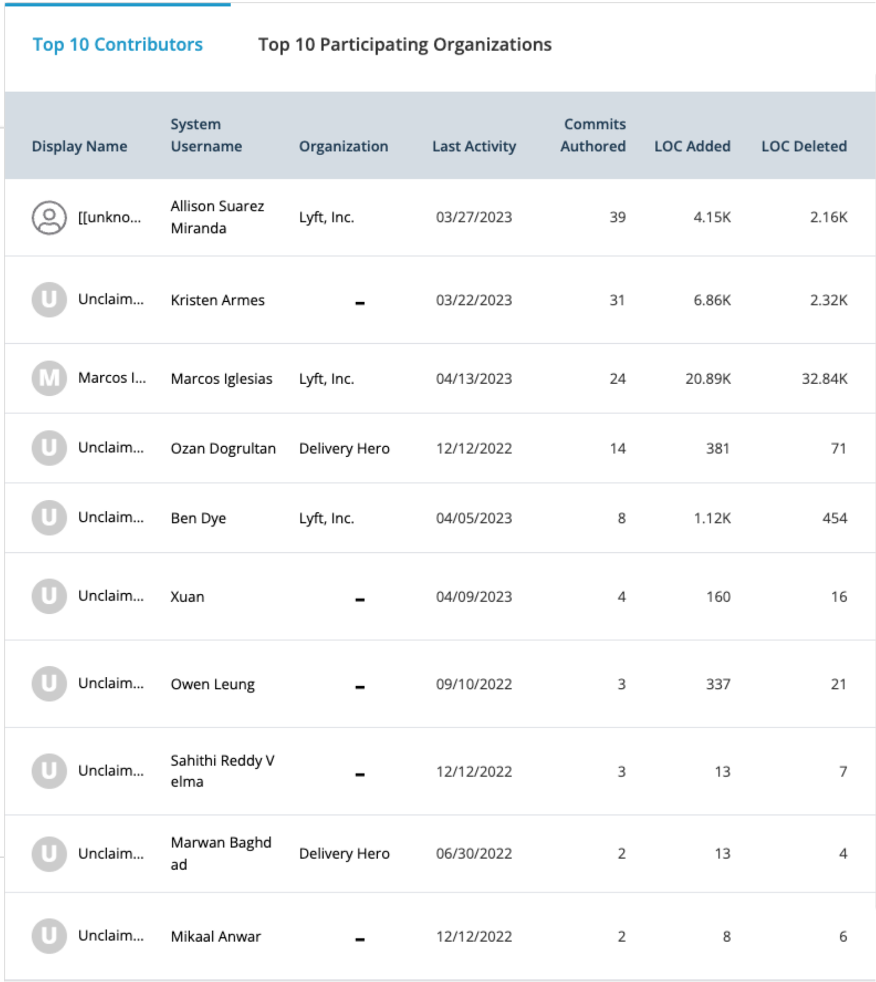
KOREA ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY · 한국과학기술원 ·
KAIST
1971

Contributions

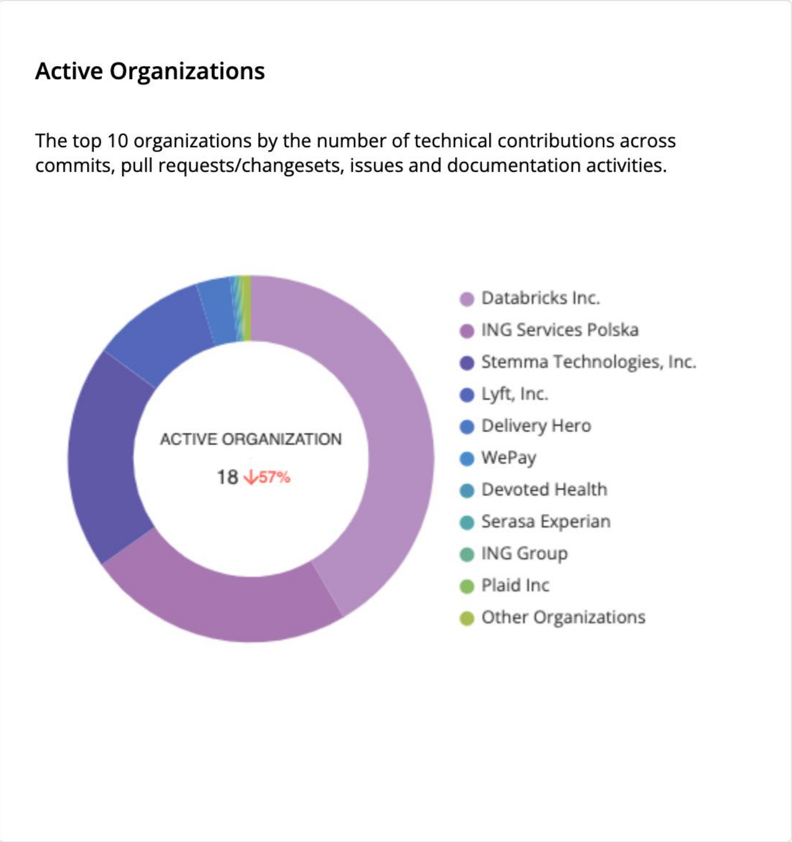


Technical Contributions

TOTAL LOC 1.14M	COMMITS 2.84K
CONTRIBUTORS 131	PULL REQUESTS 3.00K
PULL REQUESTS MERGED 2.46K	MERGED LEAD TIME 3.52 Days
ISSUES 369	ISSUES RES. TIME 193.85 Days



Display Name	System Username	Organization	Last Activity	Commits Authored	LOC Added	LOC Deleted
[[unkno...	Allison Suarez Miranda	Lyft, Inc.	03/27/2023	39	4.15K	2.16K
Unclaim...	Kristen Armes	-	03/22/2023	31	6.86K	2.32K
Marcos I...	Marcos Iglesias	Lyft, Inc.	04/13/2023	24	20.89K	32.84K
Unclaim...	Ozan Dogrultan	Delivery Hero	12/12/2022	14	381	71
Unclaim...	Ben Dye	Lyft, Inc.	04/05/2023	8	1.12K	454
Unclaim...	Xuan	-	04/09/2023	4	160	16
Unclaim...	Owen Leung	-	09/10/2022	3	337	21
Unclaim...	Sahithi Reddy Velma	-	12/12/2022	3	13	7
Unclaim...	Marwan Baghdad	Delivery Hero	06/30/2022	2	13	4
Unclaim...	Mikaal Anwar	-	12/12/2022	2	8	6



Active Organizations

The top 10 organizations by the number of technical contributions across commits, pull requests/changesets, issues and documentation activities.

Key Achievements in the past year

Overall

- Lyft team has continued to maintain Amundsen with a reduced headcount
- At least 38 companies are officially using Amundsen
- Cohesive frontend strategy to set a unified direction for future contributors

Technical

- Improved search service with fuzzy search, stemming, usage based rankings, multivalue filters
- Revamped Neo4j metadata publisher to increase publishing speeds by 97%
- Added support for newer Neo4j versions

Community

- Slack community has reached over 3k members
- 3.9k stars on Github

Areas the project could use help on

- Onboarding process for users and contributors
- Frontend migrations and other challenges
- Lack of contributors to give technical support to users
- Limited contributions from other maintainers outside of Lyft

TAC Open Discussion

LF AI & DATA

LakeSoul Introduction

A Cloud-native Realtime LakeHouse Framework

https://github.com/meta-soul/LakeSoul

Amir Banifatemi chenxu@dmetasoul.com

5/4/2023

LF AI & DATA

Why donate to LF AI & Data

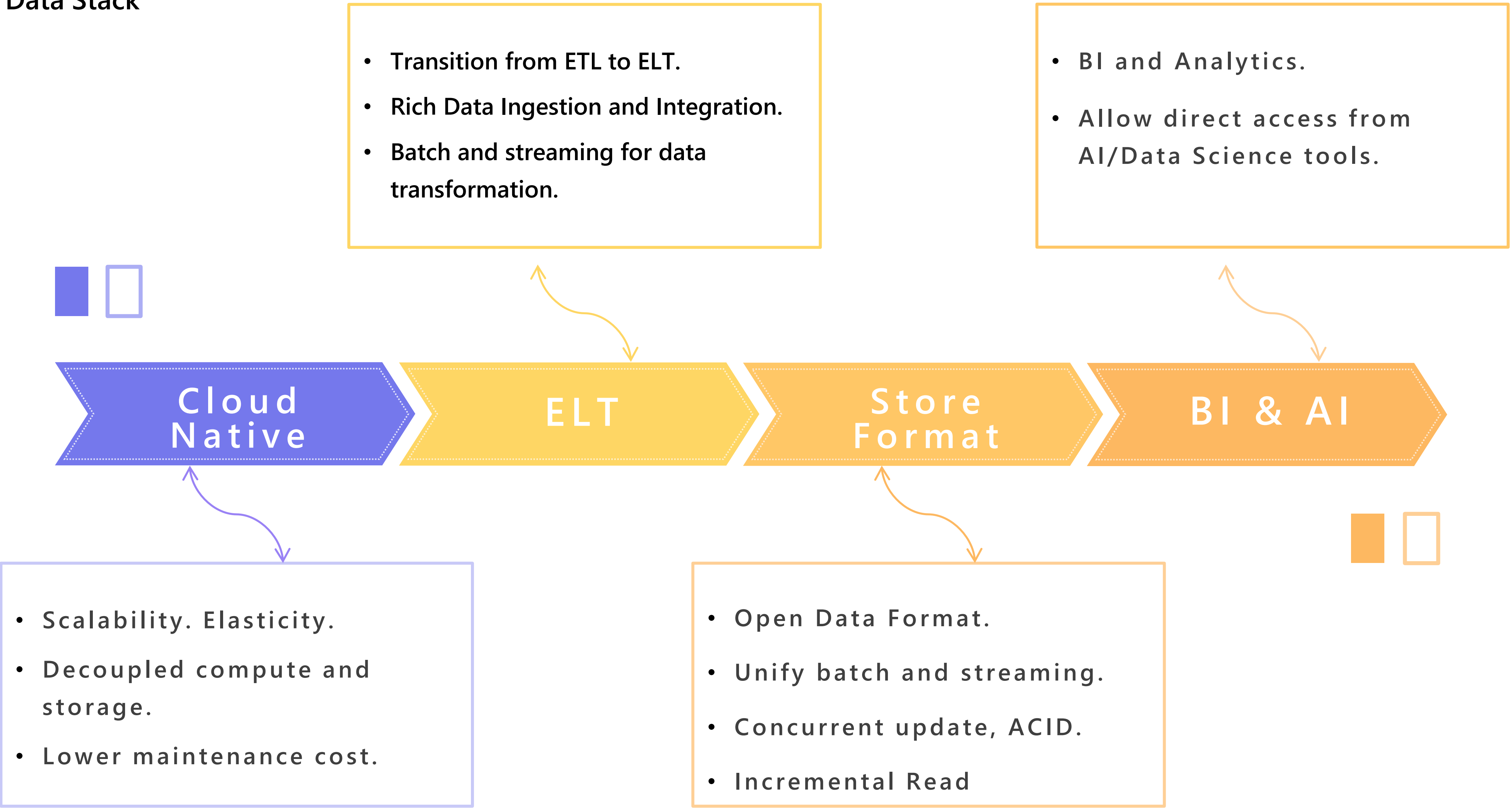
- Neutral Holding Ground
  - Vendor-neutral, Not for profit
- Growing Community
  - Increase users by outreach and involving through the foundation
  - Increase contributors from developer users
  - Collaboration with other projects in the foundation
- Open Governance Model
  - Open governance + open source license
  - Neutral management of project by the foundation
  - Instill trust in contributors and adopters in the management of the project

Company Profile

- We are DMetaSoul
    - A startup based in Beijing
    - Building better data & intelligence infrastructure
    - We believe opensource is the key!

Background

Modern Data Stack



Cloud Native

ELT

Store Format

BI & AI

- Scalability. Elasticity.
- Decoupled compute and storage.
- Lower maintenance cost.

- Transition from ETL to ELT.
- Rich Data Ingestion and Integration.
- Batch and streaming for data transformation.

- Open Data Format.
- Unify batch and streaming.
- Concurrent update, ACID.
- Incremental Read

- BI and Analytics.
- Allow direct access from AI/Data Science tools.

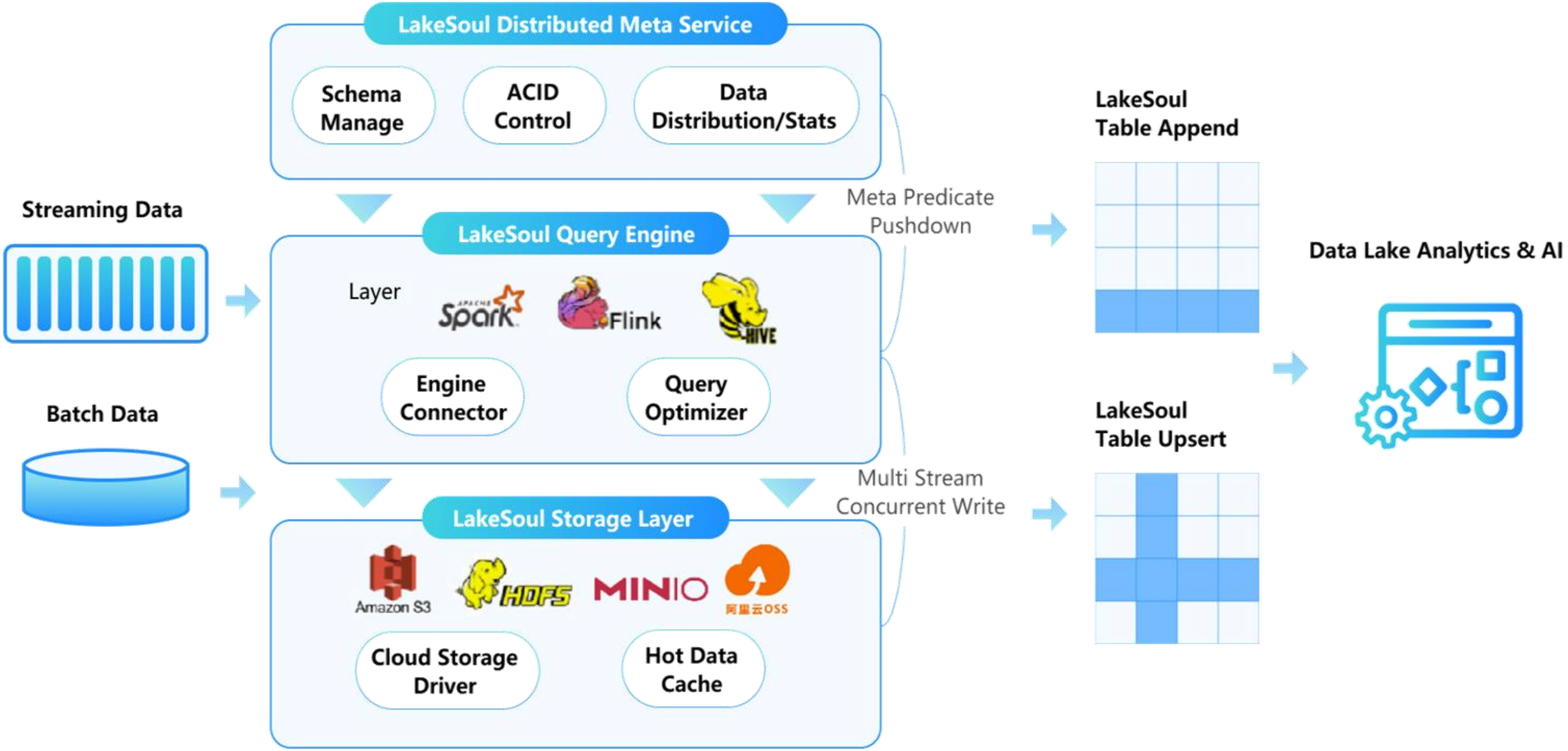
04MAY2023

LakeSoul’s Positioning

Cloud Native Lakehouse Framework with Unified Batch and Streaming Support

- Goals
    - Cloud first, without dedicated storage, and optimizations for object store
    - Centralized metadata management, ACID, concurrent upsert and snapshot read
    - Streaming data ingestion and incremental pipeline
    - Make data analytics and AI on data lake more efficient and easy
- Non Goals
    - Create a new compute engine
    - Create a new file format
    - Optimize for point update or query

LakeSoul Architectural Overview



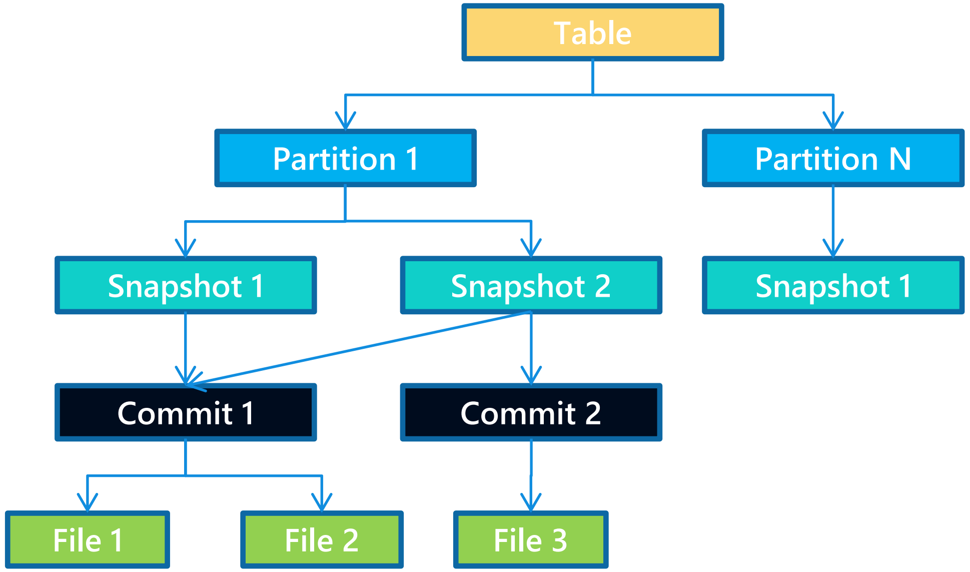
04MAY2023

Data Model and Metadata

LF AI & DATA

Data Modeling

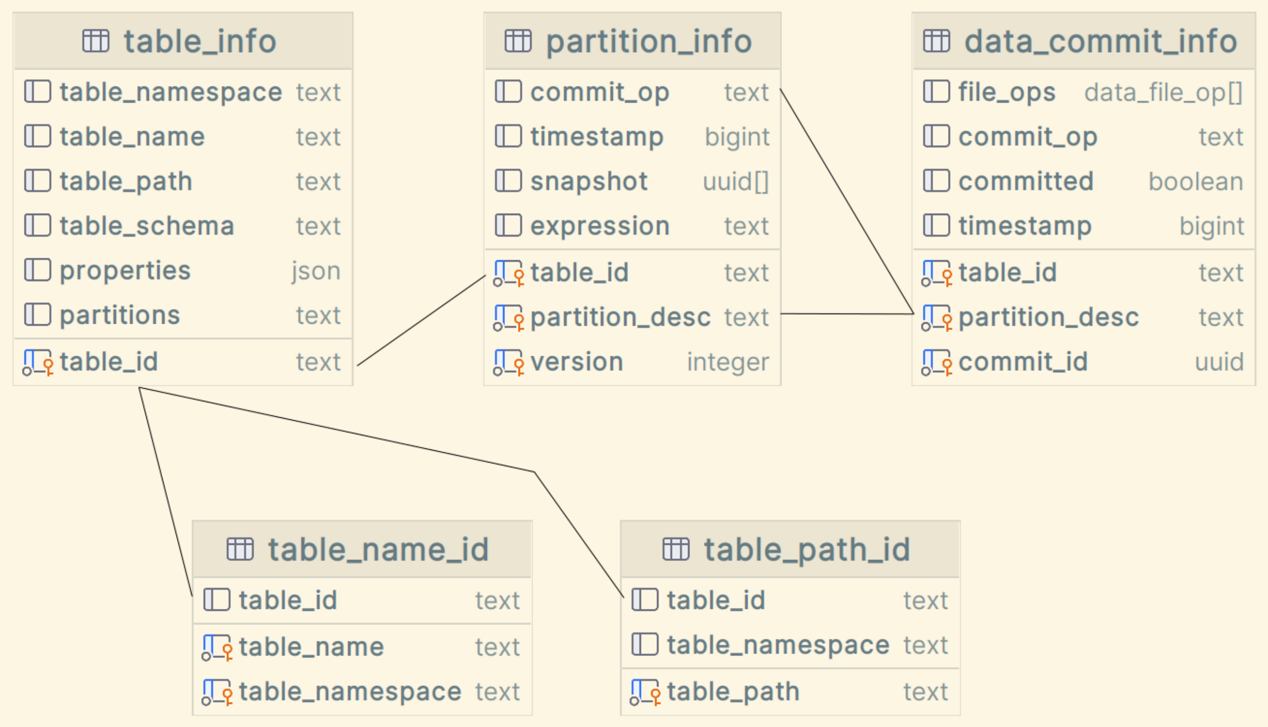
- Physical Data
  - Files are stored physically with Parquet format
  - Table could optionally have primary key constraint
    - Files are hash bucketed (with a predefined bucket number), and each upserted file is sorted by PKs
  - Table could have multi-level range partitions
- Meta Data
  - Commit: Files sequence with add/delete ops
  - Snapshot: Commits sequence with commit types (Append, Merge, Compaction, Update)
  - Version: Monotonic increasing number that identifies a snapshot and its timestamp



Centralized Metadata Management

- Centralized metadata management through PostgreSQL
  - Concurrent ACID via PG's transaction
  - Two-phase commit protocol
  - Fine-grained write conflicts resolving
  - Trigger function in PLSQL for event publish
- PG is generally available on most of the cloud vendors
- Java wrapper and Spark/Flink's Catalog interface implementations

PostgreSQL



table_info	
table_namespace	text
table_name	text
table_path	text
table_schema	text
properties	json
partitions	text
table_id	text

partition_info	
commit_op	text
timestamp	bigint
snapshot	uuid[]
expression	text
table_id	text
partition_desc	text
version	integer

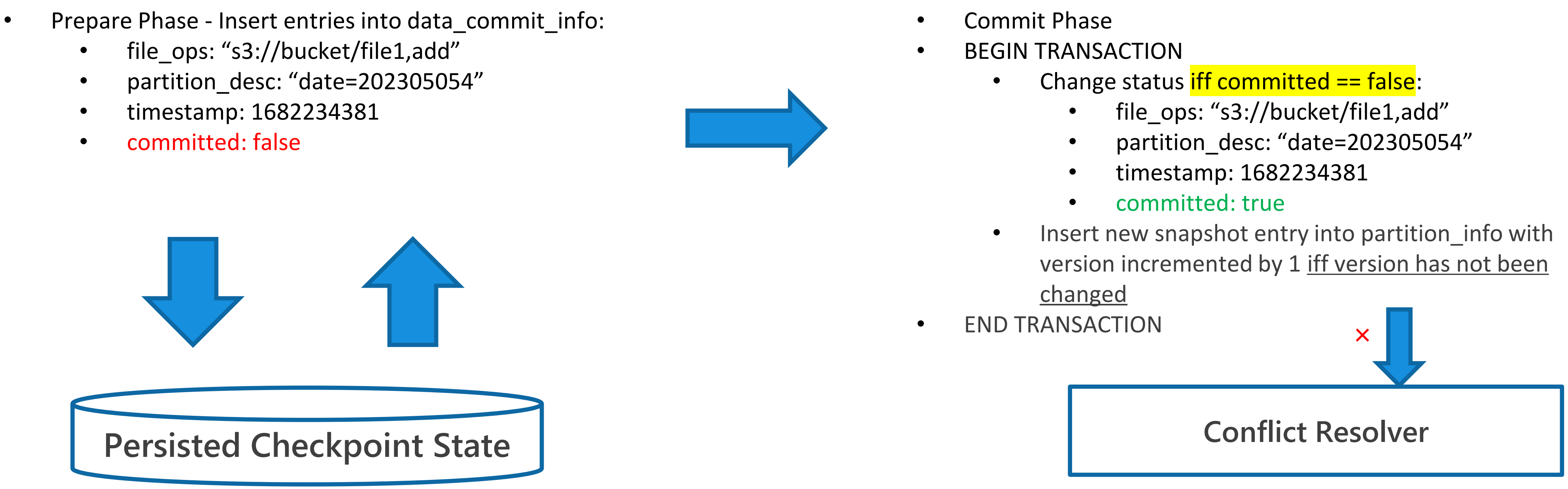
data_commit_info	
file_ops	data_file_op[]
commit_op	text
committed	boolean
timestamp	bigint
table_id	text
partition_desc	text
commit_id	uuid

table_name_id	
table_id	text
table_name	text
table_namespace	text

table_path_id	
table_id	text
table_namespace	text
table_path	text

Centralized Metadata Management

- Two-phase Commit Protocol
  - Executed during batch write or stream checkpoint in Spark/Flink



- Prepare Phase - Insert entries into data_commit_info:
  - file_ops: “s3://bucket/file1,add”
  - partition_desc: “date=202305054”
  - timestamp: 1682234381
  - committed: false

Persisted Checkpoint State

- Commit Phase
- BEGIN TRANSACTION
  - Change status iff committed == false:
    - file_ops: “s3://bucket/file1,add”
    - partition_desc: “date=202305054”
    - timestamp: 1682234381
    - committed: true
  - Insert new snapshot entry into partition_info with version incremented by 1 iff version has not been changed
- END TRANSACTION

×

Conflict Resolver

Centralized Metadata Management

- Fine-grained write conflict resolving with PG’s transaction
  - Retry: Compatible write but version changed, retry with newest version + 1
    - Concurrent Append, Merge
  - Reorder: Create a new snapshot with current commit in the middle
    - Concurrent compaction or update with no other unresolvable conflict
  - Concurrent Updates are unresolvable and fail

Operation	Append	Merge	Compaction	Update
Append	Retry	X	Retry	Retry
Merge	X	Retry	Reorder	Retry
Compaction	Reorder	Reorder	Ignore	Ignore
Update	Reorder	Reorder	Overwrite	Fail

- Guaranteed atomicity while improving concurrency

Centralized Metadata Management

- Auto Schema Evolution
  - Automatically update schema during write
  - Enabling schema change on the fly (without stop-the-world DDL operation)
  - Automatic read schema reconciliation
    - Add Column: Old data padded with null during read
    - Drop Column: Old data’s column filtered out during read
- Snapshot Read, Rollback and Cleanup
  - Each snapshot is associated with a UTC timestamp
  - Read newest snapshot by default
  - APIs to access older snapshot with human-readable timestamp string

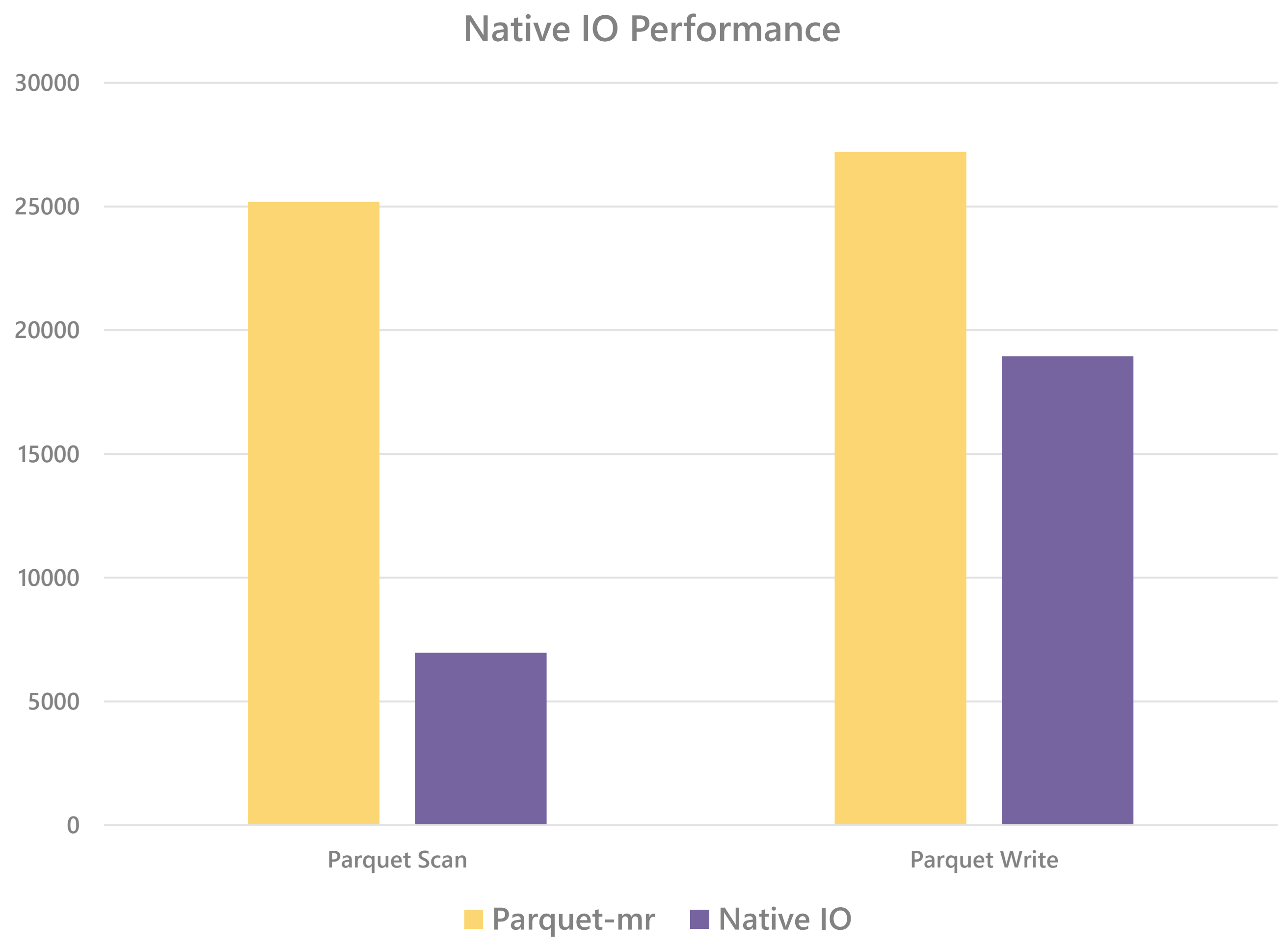
Native IO Layer

Design Principles:

1. Encapsulate read/write logics for upsert and merge on read
   1. Simple interfaces for reading/writing parquet files to/from object storage and HDFS
2. Easier integration
   1. Integrate with various data&ai compute engines
   2. Provide vectorized reader & writer if the engine needs
   3. Provide C, Java, Python wrappers
3. Cloud native
   1. Optimize for high latency r/w
   2. Limit cpu/memory usage

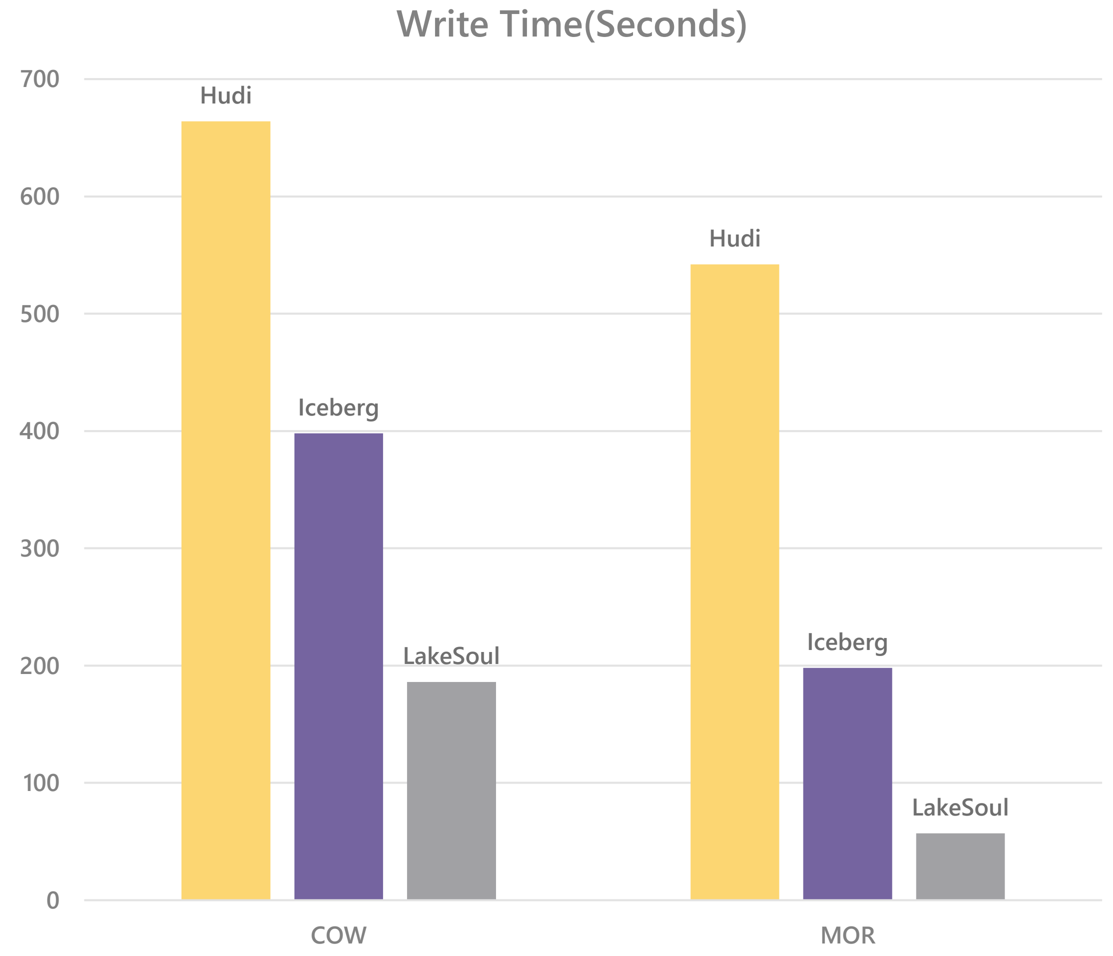
Implementation:

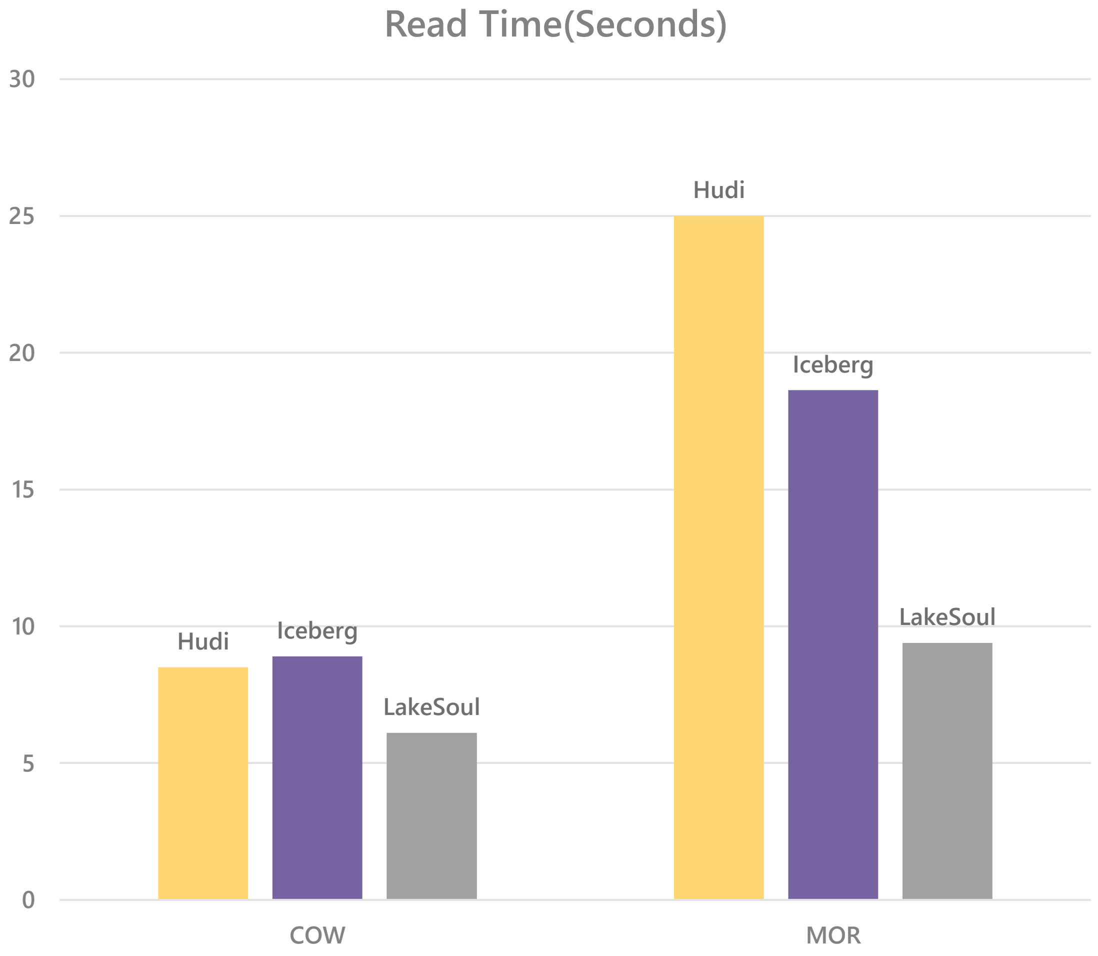
1. Async reader, writer in Rust, with arrow-rs and arrow-dataFusion
   1. Apache Arrow Recordbatch as memory format
   2. Async Writer: async sort and multi-part upload in background IO threads
   3. Async Reader: Sorted merge from async file streams with parquet row group prefetch and large request splitting
2. C interface and Java/Python wrappers through jnr-ffi/ctypes
3. Spark DataSource V2 & Flink DynamicTableFactory implementations for both batch and stream



Benchmark source code available at:
https://github.com/meta-soul/LakeSoul/tree/main/lakesoul-spark/src/test/scala/org/apache/spark/sql/lakesoul/benchmark/io

04MAY2023





Benchmark data and source code available at:
https://github.com/meta-soul/ccf-bdci2022-datalake-contest-examples/tree/mor
https://github.com/meta-soul/ccf-bdci2022-datalake-contest-examples/tree/cow

Benchmark settings:

- Environment: Spark 3.3.1, 4 cpu 16G, OpenJDK 11
- Write 10 millions rows initially, then upsert 10 times for 2 millions rows each (with 1 million existing PKs each)
- Merge on Read without compaction

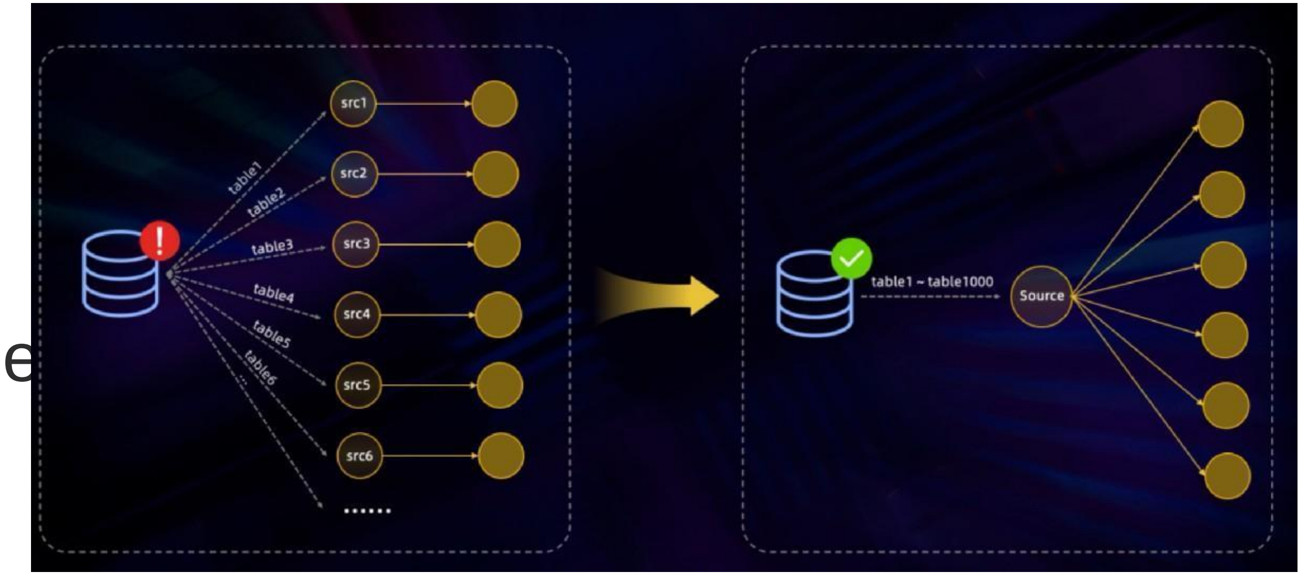
04MAY2023

Streaming Pipeline

LF AI & DATA

Streaming Data Ingestion

- Synchronize multiple tables from RDBMS (MySQL etc.) and multiple topics from message queue (Kafka)
  - In ONE Flink/Spark stream job
  - CDC stream ingestion
  - Auto new table/topic discovery
  - Auto schema change sync
  - End-to-end exactly once guarantee

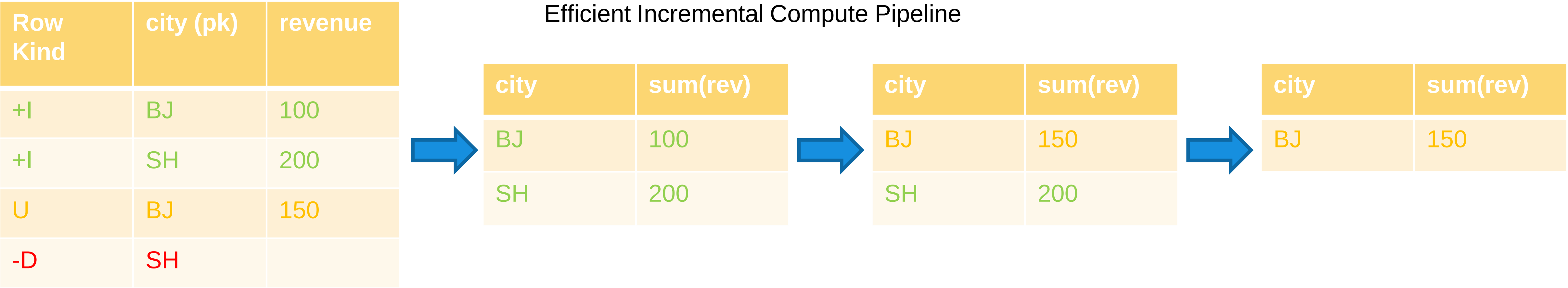


- Native Support for Changelog Format
  - Added a “row_kind” column in storage format
  - “row_kind” column’s enum values: “update”, “insert”, “delete”
  - Sink Debezium/Flink CDC streams, etc. into LakeSoul’s changelog format
  - Incremental and continuous read from LakeSoul table as changlog stream source

```
INSERT INTO lakesoul_table SELECT * FROM mysql_cdc_stream;

SELECT sum(revenue) FROM lakesoul_table
/*+ OPTIONS('readstarttime'='2023-04-21 10:00:00','readtype'='incremental')*/
GROUP BY city;
```

Efficient Incremental Compute Pipeline



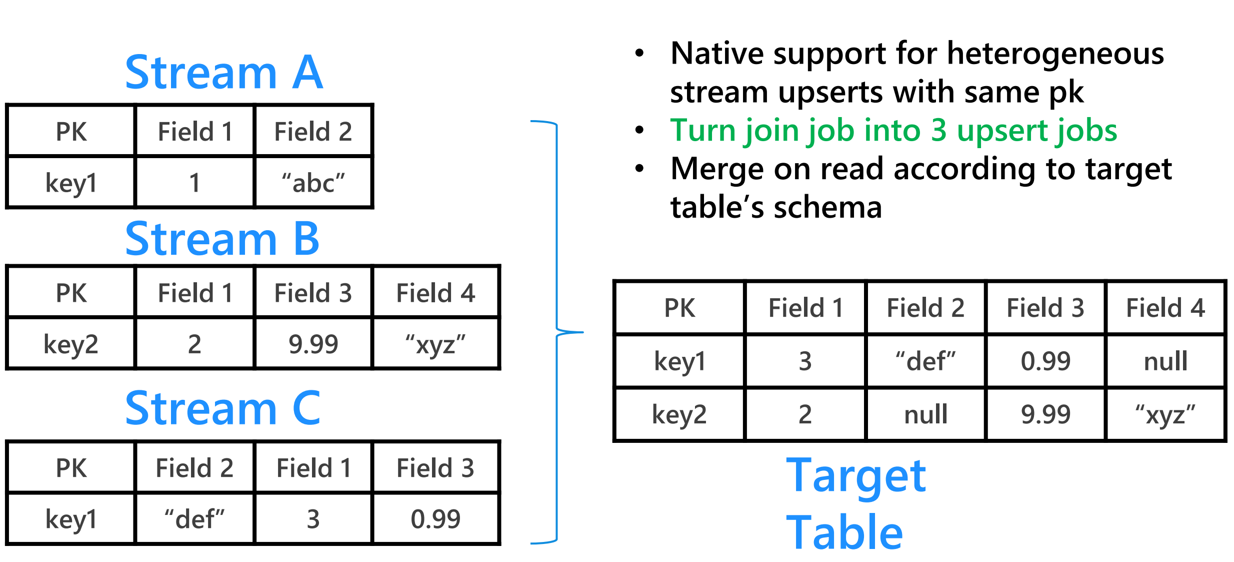
Row Kind	city (pk)	revenue
+I	BJ	100
+I	SH	200
U	BJ	150
-D	SH	

city	sum(rev)
BJ	100
SH	200

city	sum(rev)
BJ	150
SH	200

city	sum(rev)
BJ	150

- Multi streams join without engine’s state
  - Reduce maintenance overhead of large stateful stream job
  - Reduce compute overhead of full join among large tables
  - Achieve higher throughput with lower latency



Stream A

PK	Field 1	Field 2
key1	1	“abc”

Stream B

PK	Field 1	Field 3	Field 4
key2	2	9.99	“xyz”

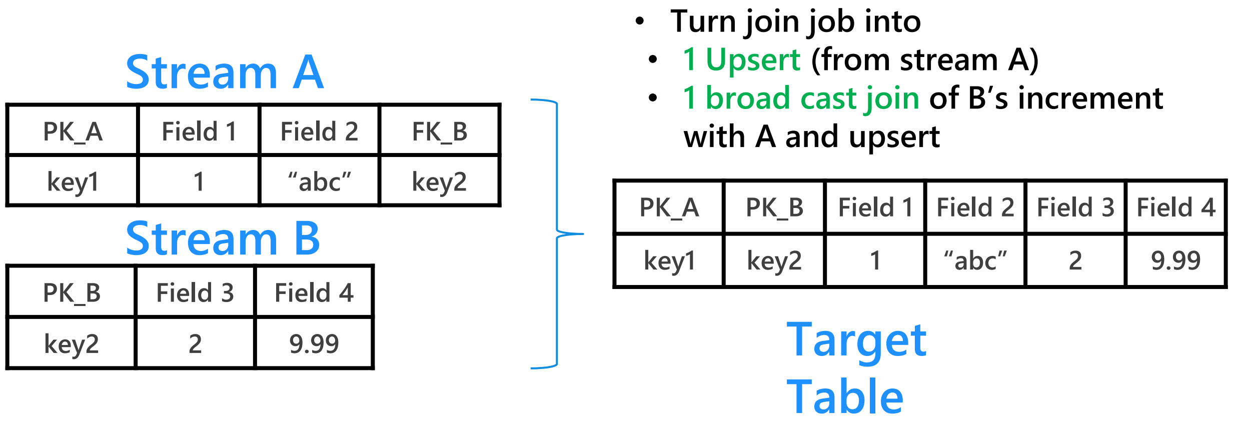
Stream C

PK	Field 2	Field 1	Field 3
key1	“def”	3	0.99

- Native support for heterogeneous stream upserts with same pk
- Turn join job into 3 upsert jobs
- Merge on read according to target table’s schema

PK	Field 1	Field 2	Field 3	Field 4
key1	3	“def”	0.99	null
key2	2	null	9.99	“xyz”

Target Table



Stream A

PK_A	Field 1	Field 2	FK_B
key1	1	“abc”	key2

Stream B

PK_B	Field 3	Field 4
key2	2	9.99

- Turn join job into
  - 1 Upsert (from stream A)
  - 1 broad cast join of B’s increment with A and upsert

PK_A	PK_B	Field 1	Field 2	Field 3	Field 4
key1	key2	1	“abc”	2	9.99

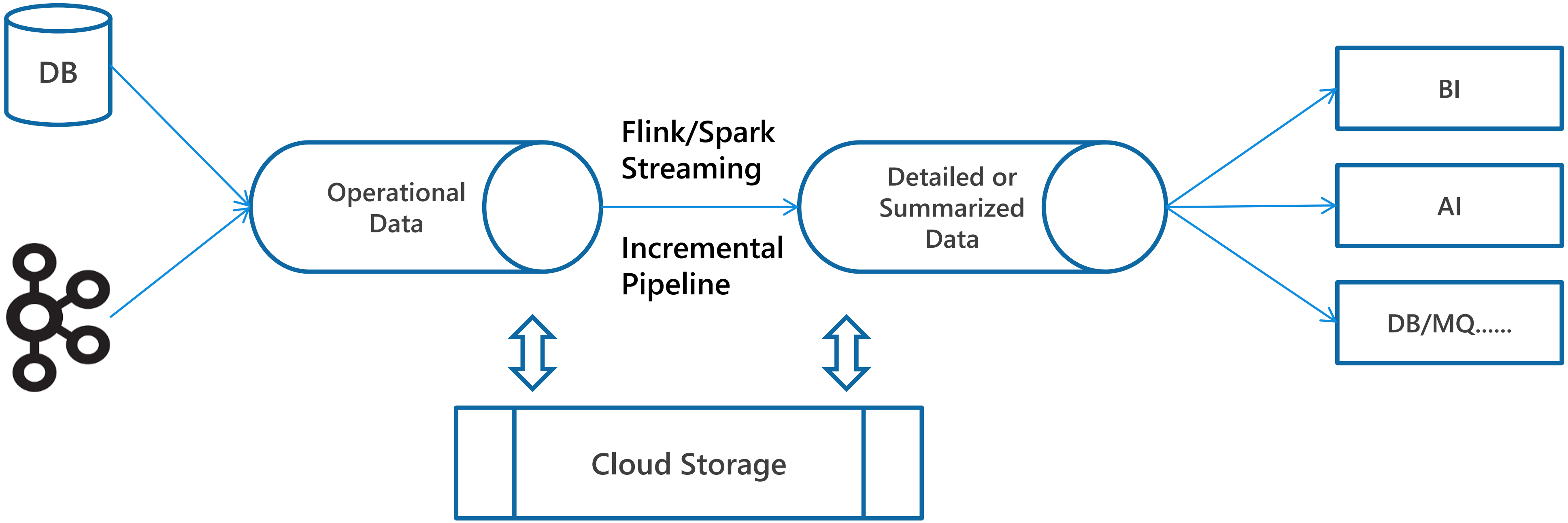
Target Table

- Rely on PG’s trigger-notify-listen mechanism
- Define a trigger function in PLSQL in PG
- Triggerred whenever new data committed and a customizable condition met (e.g. 10 commits since last compaction)
- Listen to the triggerred events for all tables and invoke compaction in one Spark job
- Auto scaling with Spark’s dynamic executor allocation

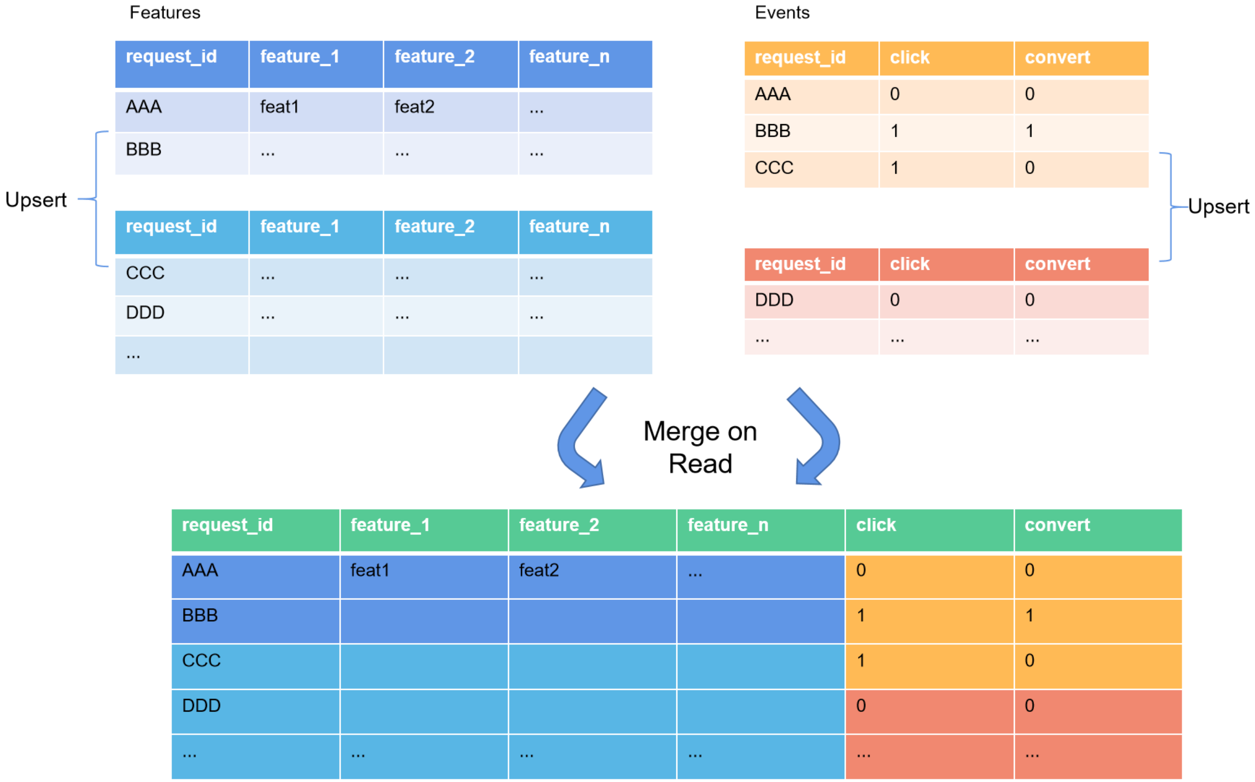
Applications

LF AI & DATA

- Build Lakehouse on LakeSoul
    - Incremental, streaming pipeline without extra time-based scheduling
    - “Unlimited” storage
    - Historical data can be accessed and updated
    - Run BI/AI on the Lakehouse
    - Pipe data to external systems



- Tabular datasets for machine learning
  - Solve decision making problems
  - Classification, forcasting, recommendations
- Use LakeSoul to build tabular datasets in real-time
  - Concat features and labels from multiple streams
  - Feed data to machine learning frameworks directly, including Spark's MLLib, Flink ML and PyTorch
  - Enable online learning by using LakeSoul table as stream source



Features

request_id	feature_1	feature_2	feature_n
AAA	feat1	feat2	...
BBB	...	...	...

request_id	feature_1	feature_2	feature_n
CCC	...	...	...
DDD	...	...	...
...			

Events

request_id	click	convert
AAA	0	0
BBB	1	1
CCC	1	0

request_id	click	convert
DDD	0	0
...	...	...

request_id	feature_1	feature_2	feature_n	click	convert
AAA	feat1	feat2	...	0	0
BBB				1	1
CCC				1	0
DDD				0	0
...	...	...	...	...	...

- Opensourced in December 2021 under Apache License V2
- 1291 Stars, 289 forks on Github
- 11 Contributors. 4 from other organizations
- Early adoptions from aviation and banking companies and one research lab

Possible Collaboration with LF AI & Data Projects

- Integrate data lineage with OpenLineage and Marquez
- Provide batch and stream source to data and feature processing projects including Sparklyr and Feast
- Build tabular training datasets for ML systems including PyTorch, Angel ML and FATE

- Data Warehousing
  - Streaming State Table
  - SQL to streaming pipeline translation
  - Data lineage
  - Built-in RBAC

- Echosystem
  - PyArrow reader
  - Presto Connector
  - More DB sources
  - Kafka Connector sink
  - Logstash sink

- Performance
  - Improve sorted stream merge speed
  - Minor compaction
  - Integrate with compute engine’s vectorization optimization
  - Local disk cache

We are requesting your support to host LakeSoul in LF AI & Data as a Sanbox Project

LF AI & DATA

Approval of LakeSoul as a Sandbox Project

Proposed Resolution:

- Approval of LakeSoul as a Sandbox project of the LF AI & Data Foundation are hereby approved.

Thank You!

LF AI & DATA

Upcoming TAC Meetings

LF AI & DATA

Upcoming TAC Meetings

- May 16 – Resilience Sandbox project, Update on the Project Lifecycle Document
- June 1 – Project review <to be scheduled>

Please note we are always open to special topics as well.

If you have a topic idea or agenda item, please send agenda topic requests to tac-general@lists.lfaidata.foundation

Open Discussion

LF AI & DATA

TAC Meeting Details

- To subscribe to the TAC Group Calendar, visit the wiki: https://wiki.lfaidata.foundation/x/cQB2 ______
- Join from PC, Mac, Linux, iOS or Android: https://zoom.us/j/430697670
- Or iPhone one-tap:
  - US: +16465588656,,430697670# or +16699006833,,430697670#
- Or Telephone:
  - Dial(for higher quality, dial a number based on your current location):
  - US: +1 646 558 8656 or +1 669 900 6833 or +1 855 880 1246 (Toll Free) or +1 877 369 0926 (Toll Free)
- Meeting ID: 430 697 670
- International numbers available: https://zoom.us/u/achYtcw7uN

Legal Notice

- The Linux Foundation, The Linux Foundation logos, and other marks that may be used herein are owned by The Linux Foundation or its affiliated entities, and are subject to The Linux Foundation's Trademark Usage Policy at https://www.linuxfoundation.org/trademark-usage, as may be modified from time to time.
- Linux is a registered trademark of Linus Torvalds. Please see the Linux Mark Institute's trademark usage page at https://lmi.linuxfoundation.org for details regarding use of this trademark.
- Some marks that may be used herein are owned by projects operating as separately incorporated entities managed by The Linux Foundation, and have their own trademarks, policies and usage guidelines.
- TWITTER, TWEET, RETWEET and the Twitter logo are trademarks of Twitter, Inc. or its affiliates.
- Facebook and the "f" logo are trademarks of Facebook or its affiliates.
- LinkedIn, the LinkedIn logo, the IN logo and InMail are registered trademarks or trademarks of LinkedIn Corporation and its affiliates in the United States and/or other countries.
- YouTube and the YouTube icon are trademarks of YouTube or its affiliates.
- All other trademarks are the property of their respective owners. Use of such marks herein does not represent affiliation with or authorization, sponsorship or approval by such owners unless otherwise expressly specified.
- The Linux Foundation is subject to other policies, including without limitation its Privacy Policy at https://www.linuxfoundation.org/privacy and its Antitrust Policy at https://www.linuxfoundation.org/antitrust-policy. each as may be modified from time to time. More information about The Linux Foundation's policies is available at https://www.linuxfoundation.org.
- Please email legal@linuxfoundation.org with any questions about The Linux Foundation's policies or the notices set forth on this slide.

LF AI & DATA

04MAY2023