LedgerDB : A Centralized Ledger Database for Universal Audit and Verification

- Ant Financial Services Group
- Alibaba Group
Terminologies

• DLT (Decentralized Ledger Technology)
• CLT (Centralized Ledger Technology)
  • CLD (Centralized Ledger Database): LedgerDB, QLDB, Oracle BC Table, ProvenDB, etc.
• Immutability: Any piece of data, once committed into the system, cannot be modified by subsequent operations and becomes permanently available.
• Verifiability: The capability of validating specific data integrity and operation proofs.
• Auditability: The capability of observing a serial of user actions and operation trails based on predefined audit rules.
  • Internal audit: an internal user of the ledger can observe and verify the authenticity of all actions.
  • External audit: an external third-party entity can observe and verify the authenticity of all actions.
Why CLD is important & valuable?

• Motivations
  • Decentralization is not proved to be indispensable for permissioned blockchain.
  • Conventional permissioned blockchain and CLD systems:
    • Low performance, storage overhead, regulatory issues, limited external auditability

• Gartner Forecast
  • Gartner Strategic Vision 2019
    Strategic Planning Assumption
    By 2021, at least 20% of projects envisioned to run on permissioned blockchains will instead run on centralized, auditable ledgers.

  • Gartner Strategic Vision 2020
    By 2021, most permissioned blockchain uses will be replaced by ledger DBMS products.
Highlight and Comparison

- LedgerDB – a ledger database that provides tamper-evidence and non-repudiation features in a centralized manner (CLD), which realizes strong auditability, high performance, and data removal support.

- Key comparisons between LedgerDB and other systems.

<table>
<thead>
<tr>
<th>System</th>
<th>Throughput (max TPS)</th>
<th>Auditability</th>
<th>Removal</th>
<th>Non-Repudiation</th>
<th>Provenance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>external</td>
<td>third party</td>
<td>peg</td>
<td>capability</td>
</tr>
<tr>
<td>LedgerDB</td>
<td>100K+</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>strong</td>
</tr>
<tr>
<td>QLDB [7]</td>
<td>1K+</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>weak</td>
</tr>
<tr>
<td>Hyperledger [6]</td>
<td>1K+</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>weak</td>
</tr>
<tr>
<td>ProvenDB [40]</td>
<td>10K+</td>
<td>x</td>
<td>Bitcoin</td>
<td>✓</td>
<td>medium</td>
</tr>
<tr>
<td>Factom [43]</td>
<td>10+</td>
<td>✓</td>
<td>Bitcoin</td>
<td>✓</td>
<td>strong</td>
</tr>
</tbody>
</table>
LedgerDB system architecture.

**Ledger master** - manage the runtime metadata of the entire cluster (e.g., status of servers and ledgers) and coordinate cluster-level events (e.g., load balance, failure recovery).

**Ledger proxy** - receive client requests and preprocesses, and then dispatch them to the corresponding ledger server.

**Ledger server** - complete the final processing of requests, and interact with underlying storage layer that stores ledger data.
LedgerDB adopts an *execute-commit-index* transaction management approach:

1. **execute** - a transaction first enters the execute phase based on its transaction type. It runs on ledger proxy for better scalability.
2. **commit** - collect multiple executed transactions, arranges them in a global order (json), and persist them to the storage system. It runs on ledger server.
3. **Index** - start on ledger server to build indexes for subsequent data retrieval and verification.
Two-way peg TSA notary journals

• A TSA journal contains a ledger snapshot (i.e., a ledger digest) and a timestamp, signed by TSA in entirety. These journals are mutually entangled between each other, which provide external auditability for timestamps.

• Two-way peg protocol: ① a ledger digest is first submitted and then signed by TSA; ② TSA journal is recorded back on ledger as a TSA journal.

• We offer T-Ledger service on Alibaba Cloud LaaS+ (Ledger-as-a-Service).
Verifiable Data Removals

- **Purge**

  A purge operation deletes a set of contiguous (obsolete) journals starting from genesis to a designated jsn on ledger.

- **Occult**

  An occult operation converts the original journal to a new one that only keeps its metadata, and retains its digest.

```
01 | DELETE FROM ledger_uri
02 | WHERE jsn < pur_json;
03 | SET TS = na, cps = CONCAT(
04 |   seqX, journal_hash, blanks)
05 | WHERE jsn = Seq
06 | OR cid = des_cid;
```
Clue – Native lineage in LedgerDB

- A clue is a user-specified label (key) that carries on business logic for data lineage.
- A typical clue use case of copyrights ledger of NCAC:

  - LedgerDB conducts a write-optimized clue index structure by a reversed clue Skiplist (cSL).
  - For clue verification, we apply a dedicated verification protocol combining a clue-counter MPT (ccMPT) \( n \) and each journal verification based on \( n \).
Evaluation – cSL & bAMT

bAMT vs. Libra accumulator

(cSL vs. RocksDB)

(a) cSL Throughput

(b) cSL Latency

(c) Throughput comparison

(d) Latency comparison
Evaluation – performance and appl

LedgerDB end-to-end performance

LedgerDB is 80X faster compared to Hyperledger Fabric in the same notarization application.
LedgerDB in Production

Federated ledger vs. permissioned blockchain

LedgerDB customer use cases

- Finance (Bank, Insurance, SCF)
- IP (copyrights)
- IoT
- Regtech
- Retail
- Internet
- MISC (Healthcare, Manufacture, Energy...)

(a) Federated ledger

12% 6% 15% 45%

(b) Permissioned blockchain
Decentralized vm-like exec is just an implementation, the soul of consensus in ledger technique is dancing with time and cryptographic theorem.

- LedgerDB

https://www.aliyun.com/product/ledgerdb domestic
https://www.alibabacloud.com/product/ledgerdb international

Thanks!