



Ex Falso SQL Quodlibet

Chris Douglas

UC Berkeley



Overview

Open table formats should be logs

Constraints and deployment rules in the log

Policy change in the log

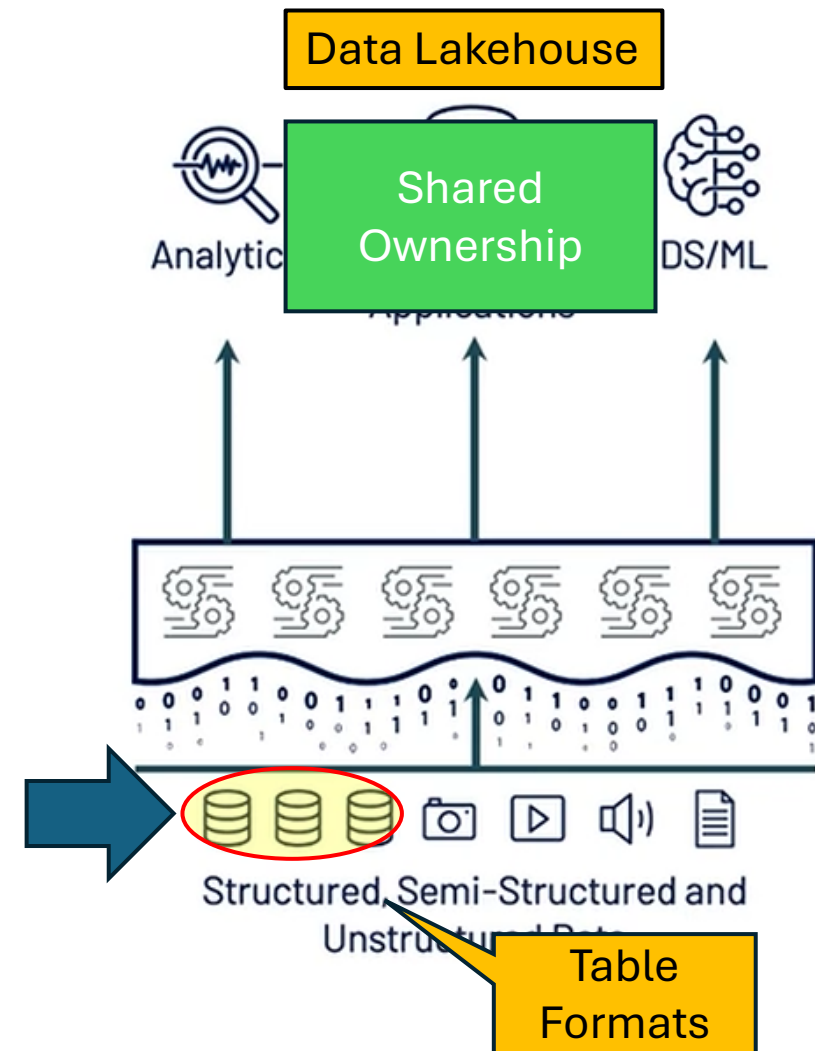
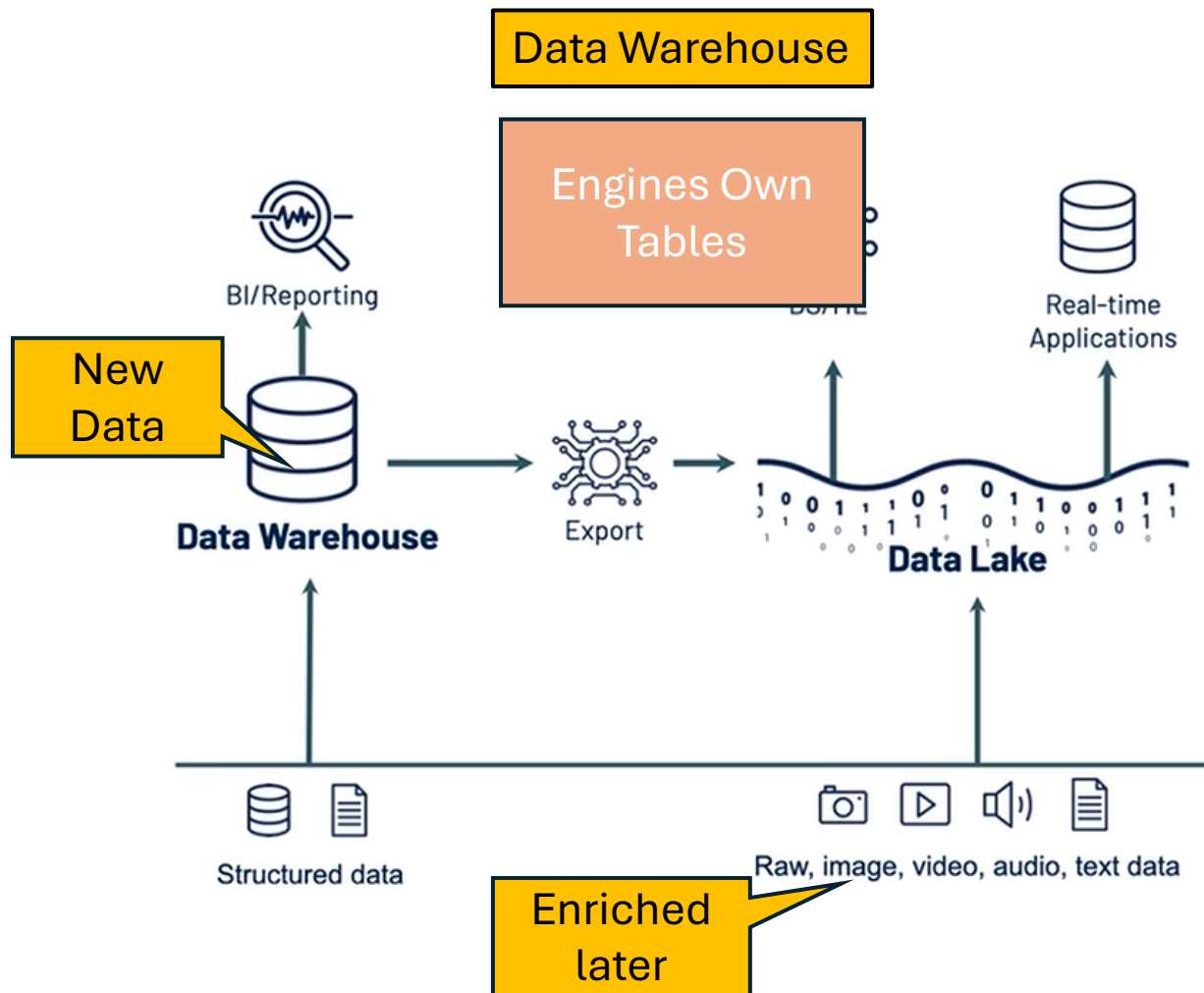
Overview

Open table formats should be logs

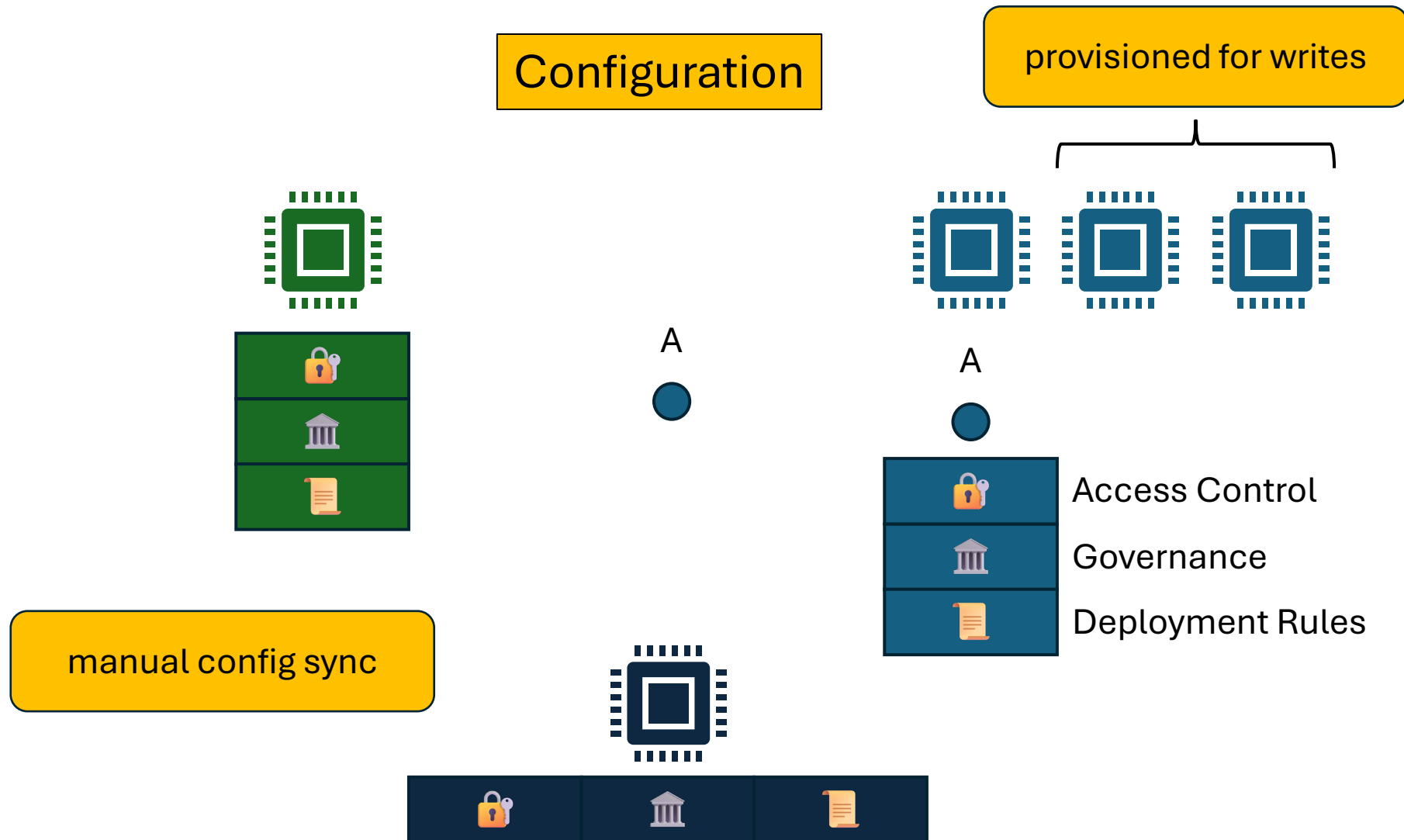
Constraints and deployment rules in the log

Policy change in the log

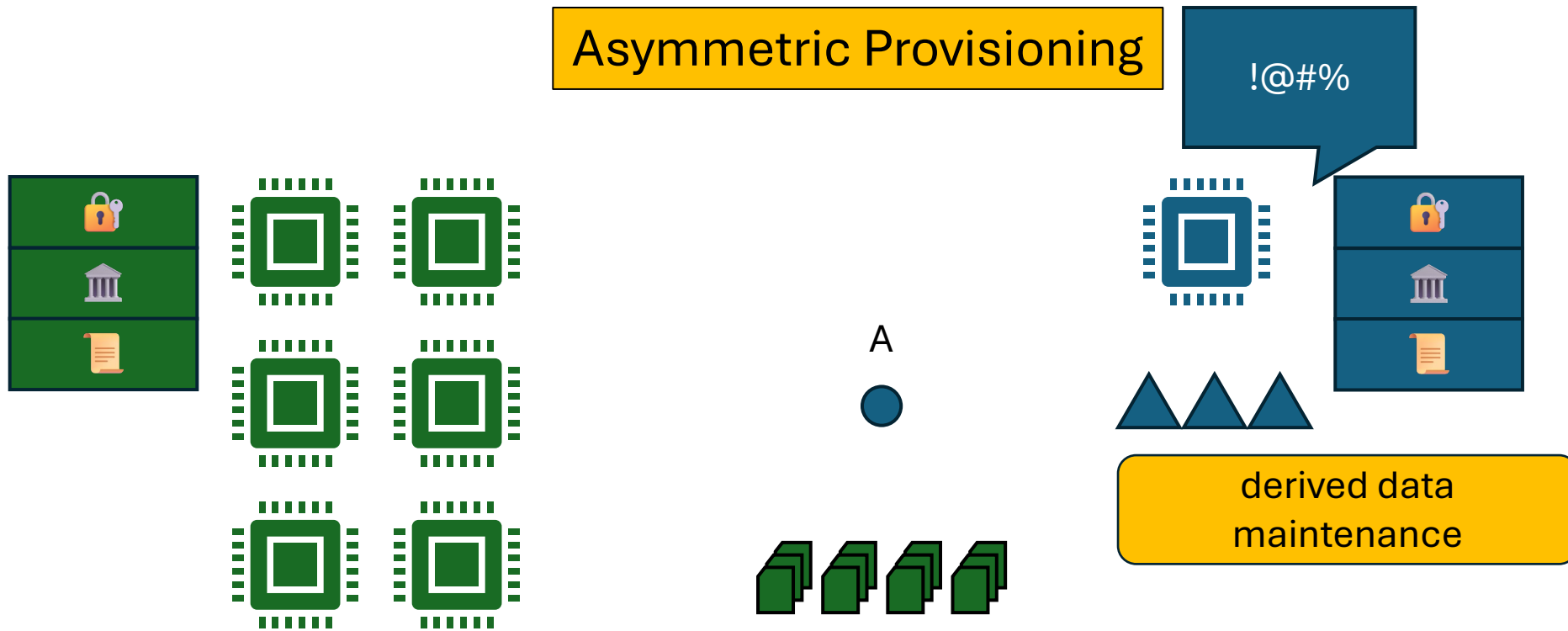
Data Lakehouse né Data Warehouse



Shared Ownership: Motivation



Shared Ownership: Motivation

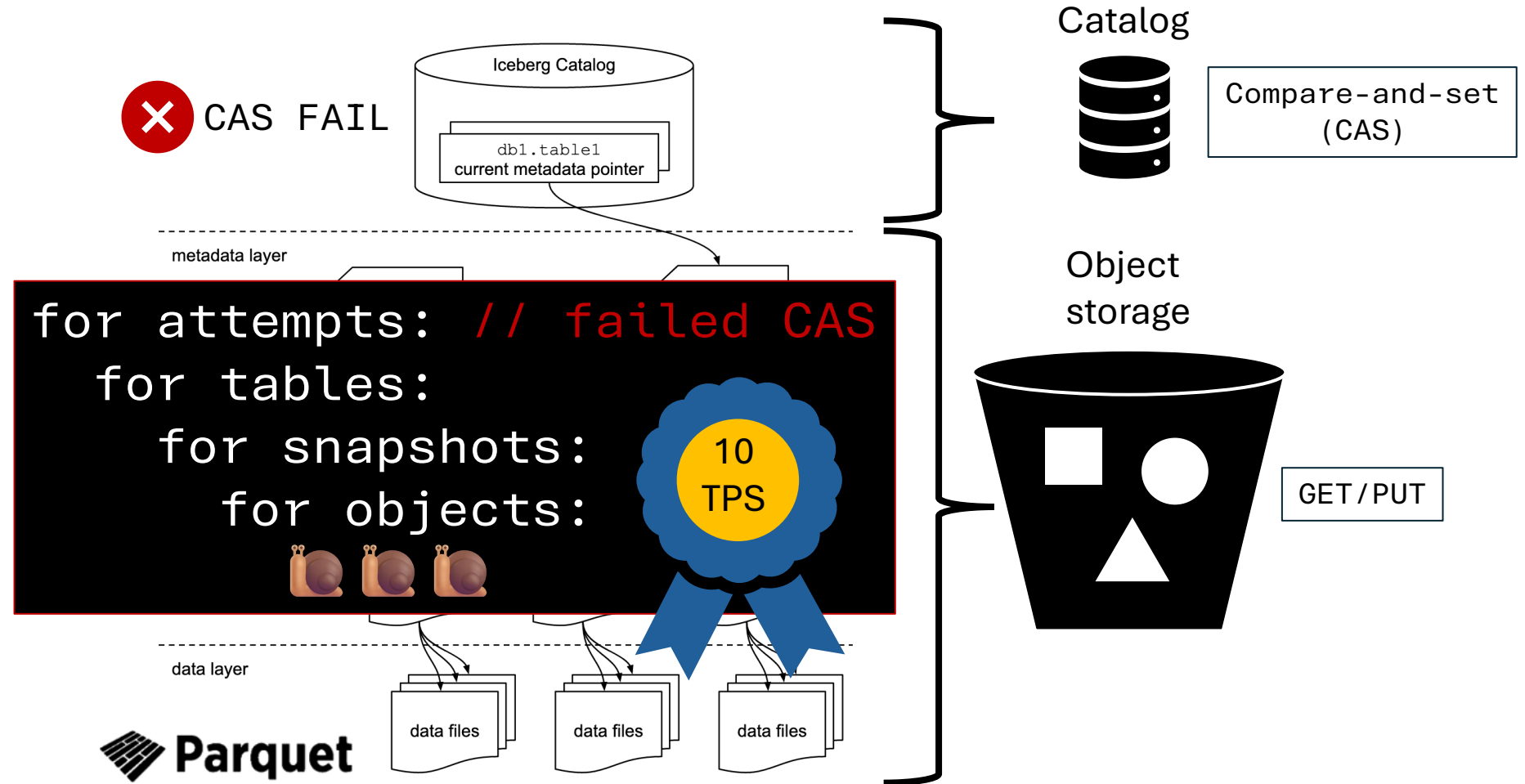


Transaction Eversion



Credit: Linda Thomas-Fowler
<https://www.flickr.com/photos/ac4lt/>

Table Formats (e.g., Apache Iceberg)



Commit Services



✖ Apache Polaris™

AWS Glue



Google Cloud



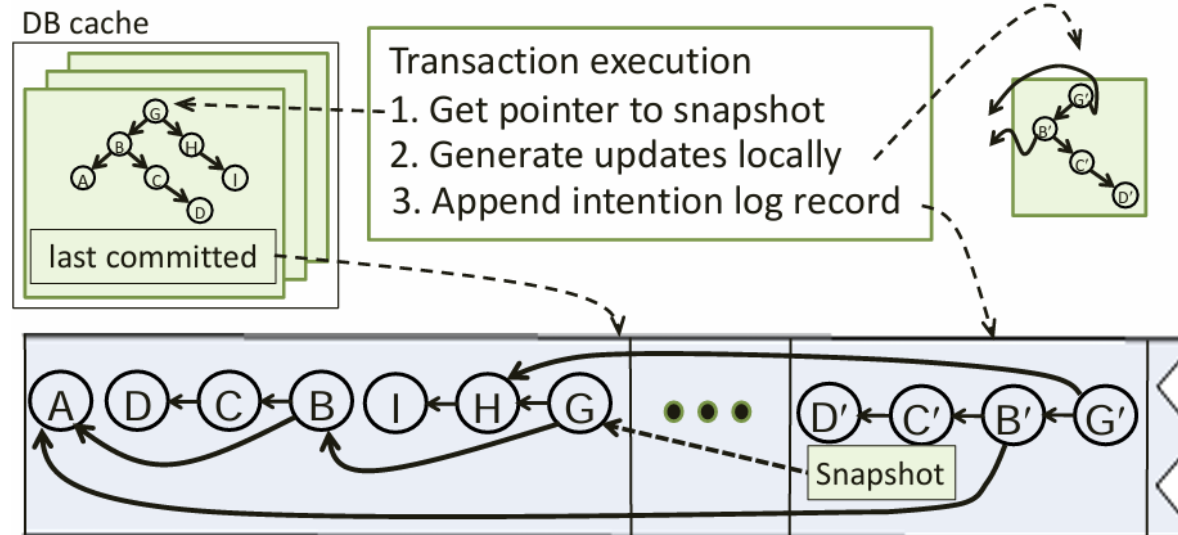
Lakekeeper

Logs Not Tables

Hyder – A Transaction Record Manager for Shared Flash
Bernstein, Reid, Das. CIDR 2011

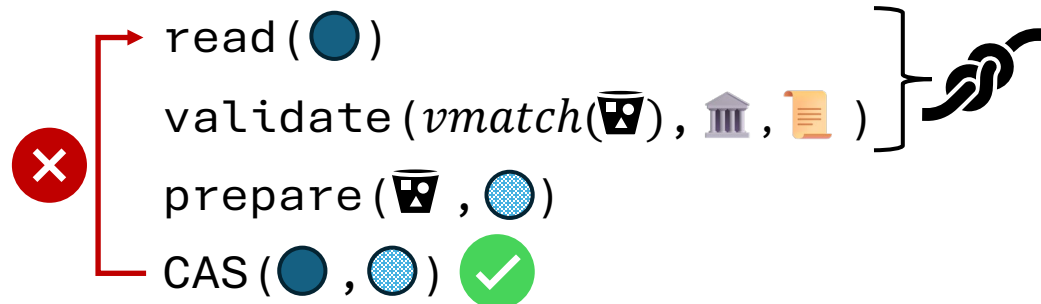
Transaction Execution

- Each server has a cache of the last committed database state
- A transaction reads a snapshot and writes an **intention log record**



Logs Not Tables

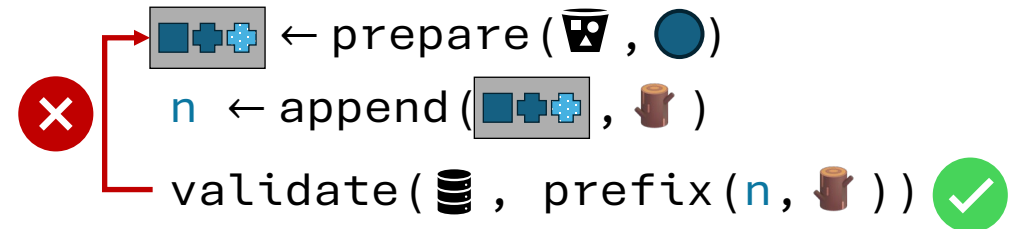
CAS Loop



SLOW

append

intention record: [box with plus signs]



FAST



Overview

Open table formats should be logs

Constraints and deployment rules in the log

Policy change in the log

What's missing? Half the DBMS Catalog!

HPTS

Read

- ✓✓ Bandwidth (object store)
- ✓ Schema
- ✓ Statistics
- ✓ Read Restrictions

Query
Planning

Parquet files

Metadata to avoid reading all of them

LPTS

Write

- ✓✓ Table Update
- ✗ Governance
- ✗ Deployment Rules
- ✗ Indexes
- ✗ Materialized Views
- ✓ Access Control

Configuration

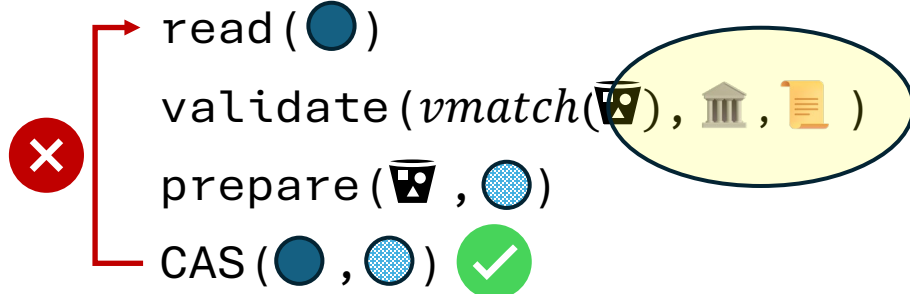
Asymmetric
Provisioning

Configuration



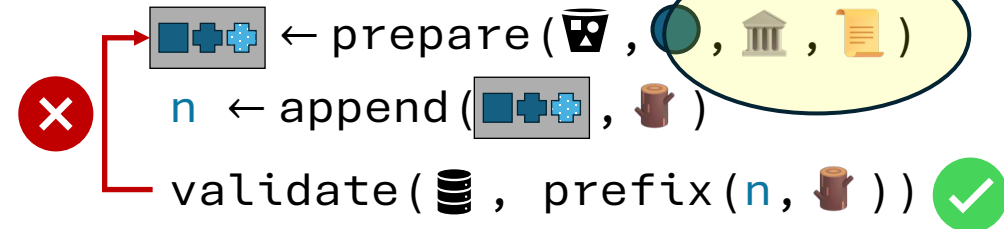
$vmatch(R) \wedge \text{🏛️} \wedge \text{📄}$

CAS Loop



append

intention record: 

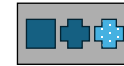


Configuration



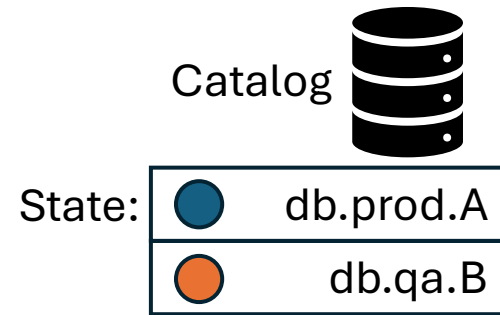
State: `% catalog state`
`ns( db.prod, -, "prod").`
`ns( db.qa, -, "qa").`
`tbl(db.prod.A,  $V_A$ , db.prod, "A").`
`tbl(db.qa.B,  $V_B$ , db.qa, "B").`

conditions evaluate
over state



payload changes state

Configuration



conditions evaluate
over state



payload changes state

Configuration



State:

	db.prod.A
	db.qa.B

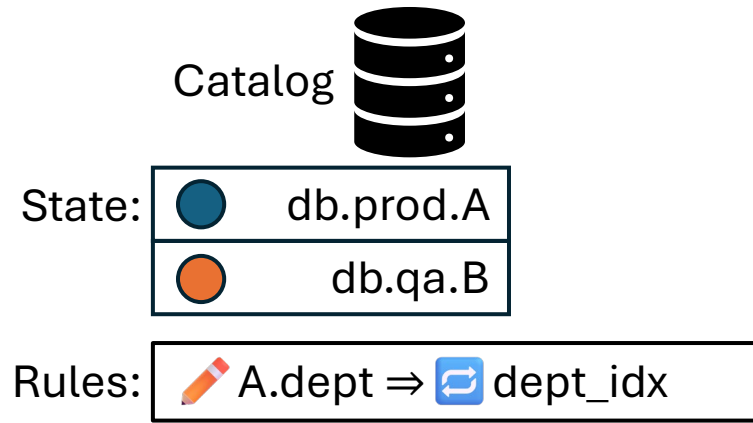
Rules:

```
% An index affected by this write
affected_index(R, Idx) :-
    writes_attr(R, Tbl, Attr),
    index(Idx, Tbl, Attr).

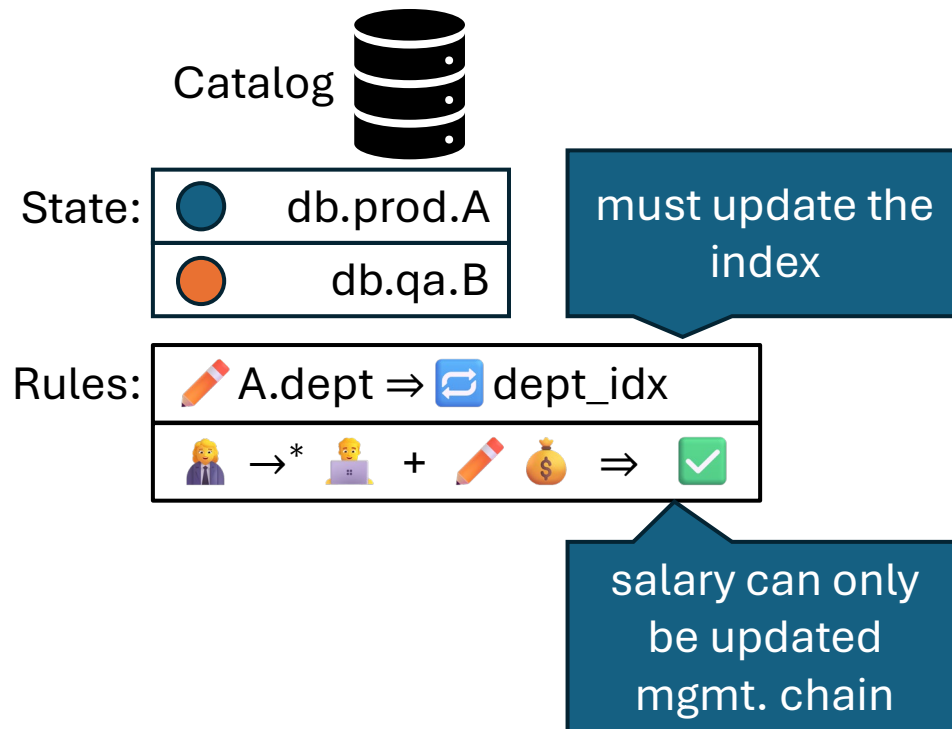
% Every affected index must be maintained
invalid(R) :-
    affected_index(R, Idx),
    not updates_index(R, Idx).

admissible(R) :- intention(R), not invalid(R).
```

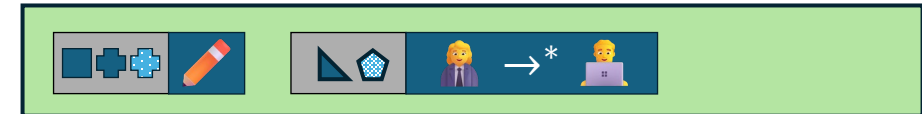
Configuration



Configuration



FIFO Queue

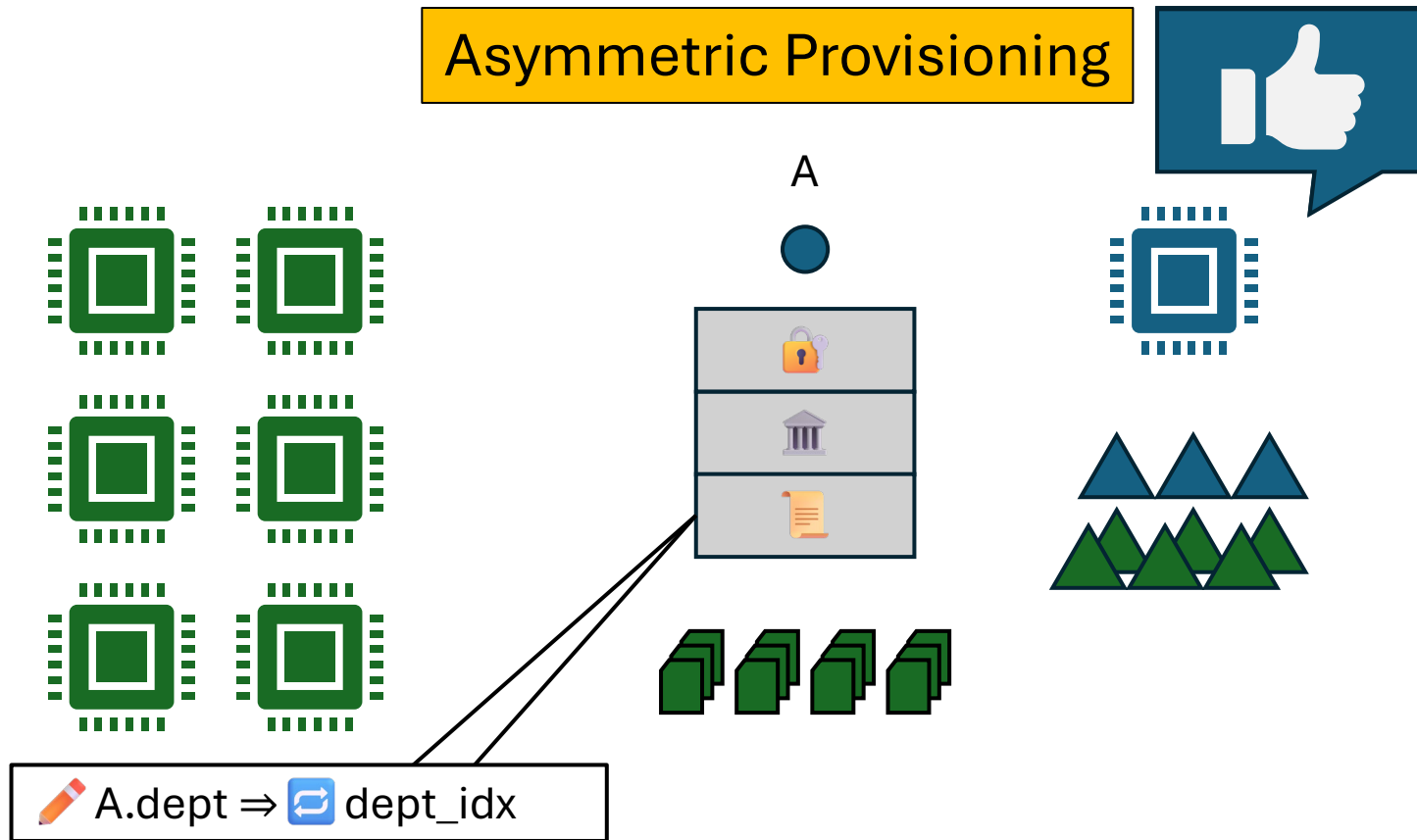


Intention records also carry a witness to the relevant rules

$vmatch(R) \wedge$  $\wedge$ 

Not only reads and writes but proofs* of invariants

Shared Ownership





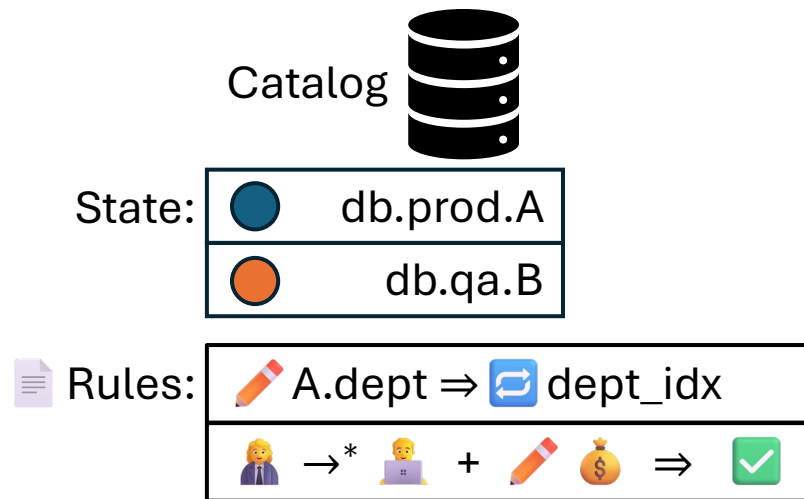
Overview

Open table formats should be logs

Constraints and deployment rules in the log

Policy change in the log

Updating Configuration



FIFO Queue



disable feature on
production tables

Maintain rules in the log

- versioned ruleset
- ordered application



Takeaways

Ex Falso Sequitur Quodlibet

From falsehood anything follows

- **Logs Not Tables**
 - Bernstein Aere Perennius
 - Materialized conditional writes
- Lower the rest of the catalog
 - Invariants + Constraints
 - Indexes + materialized views
 - Access control
- Update and enforce policy
 - Checkpoint+IR* $\Rightarrow$ IR
 - Deployment defines enforcement