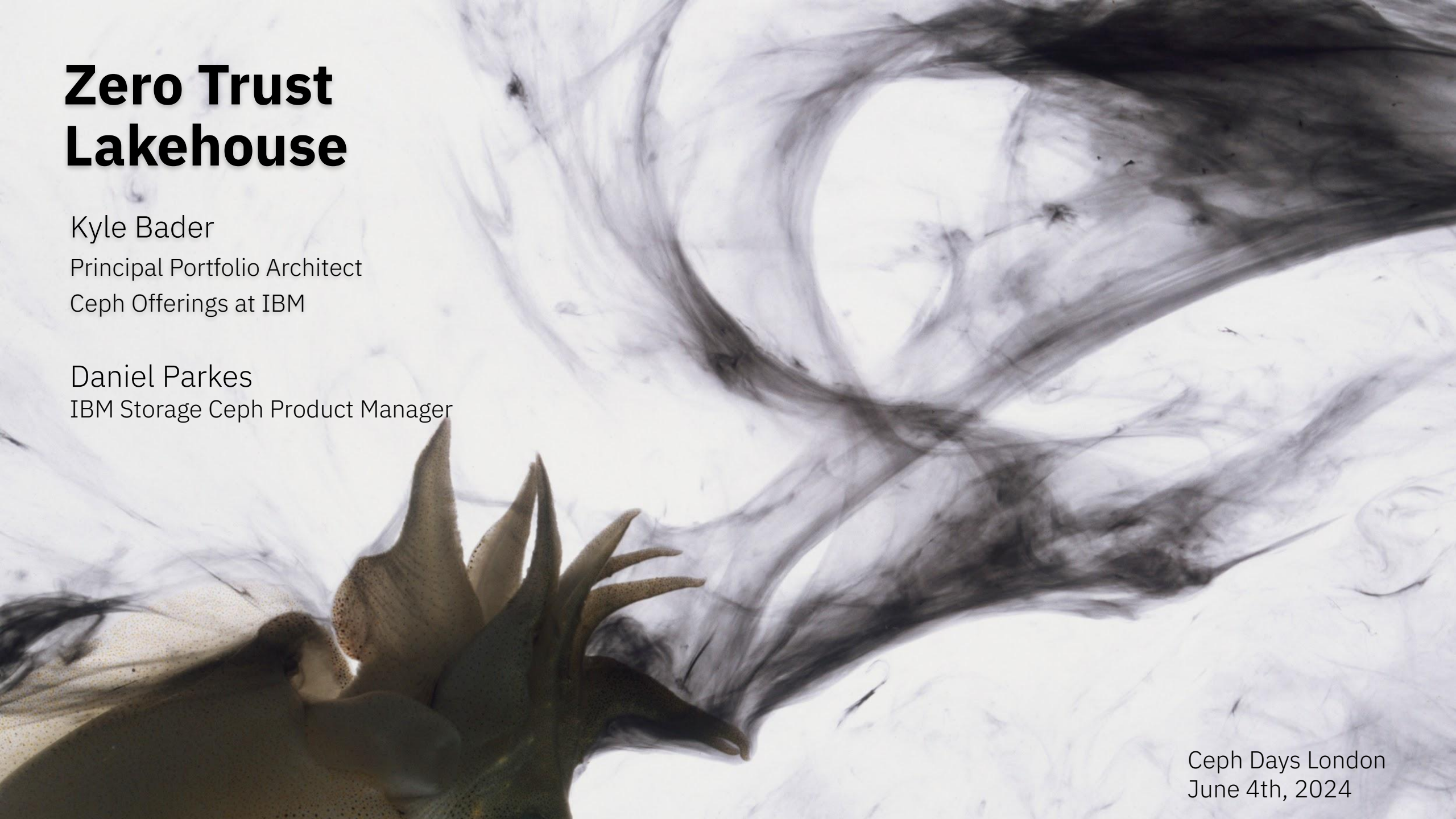
Zero Trust Lakehouse

Kyle Bader
Principal Portfolio Architect
Ceph Offerings at IBM

Daniel Parkes
IBM Storage Ceph Product Manager



Ceph Days London
June 4th, 2024

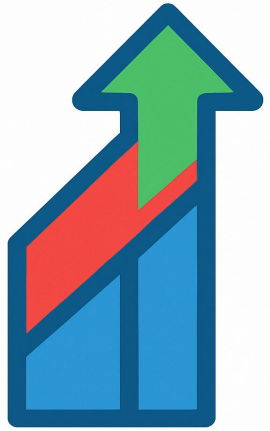
Ceph Object Storage: Foundation for Open Analytics

Industry Trend

Open table formats on object storage fuse scattered data into one unified, high-performance lake.

Object Storage

Ceph object storage is the common data platform for all structured and unstructured data



Innovation

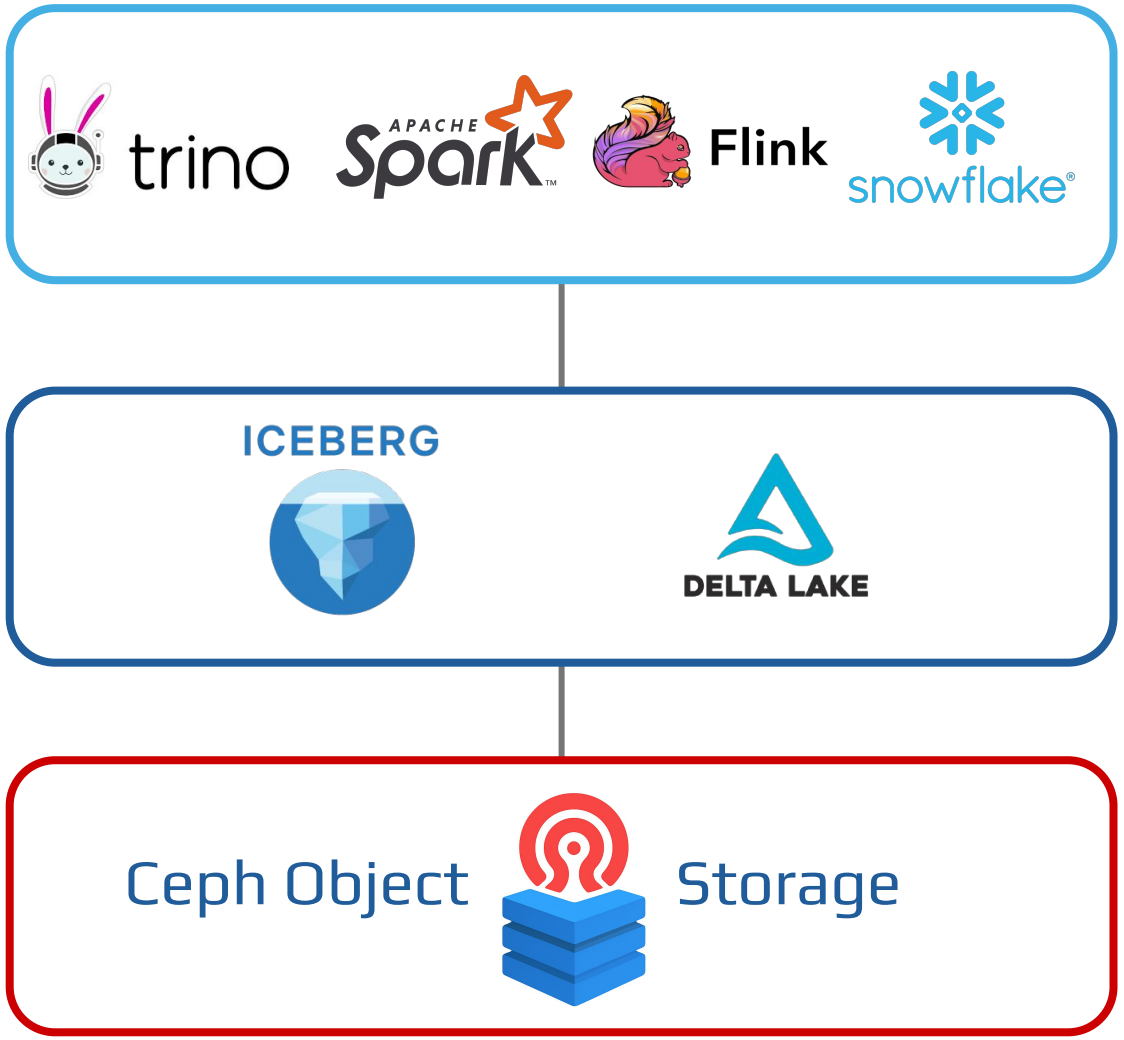
Ceph Object innovation in governance, performance, and open analytics.

Ceph Object Storage: Foundation for Open Analytics

Compute Agnostic
Pick Your Engine

Open Table Formats
Open Standards

Storage Foundation
Ceph Object



Lakehouse

“In general, a modern cloud analytics architecture separates compute from storage, with key capabilities pushed to the durable storage component to address enterprise customer requirements, such as governance, security, metadata management, and performance.”

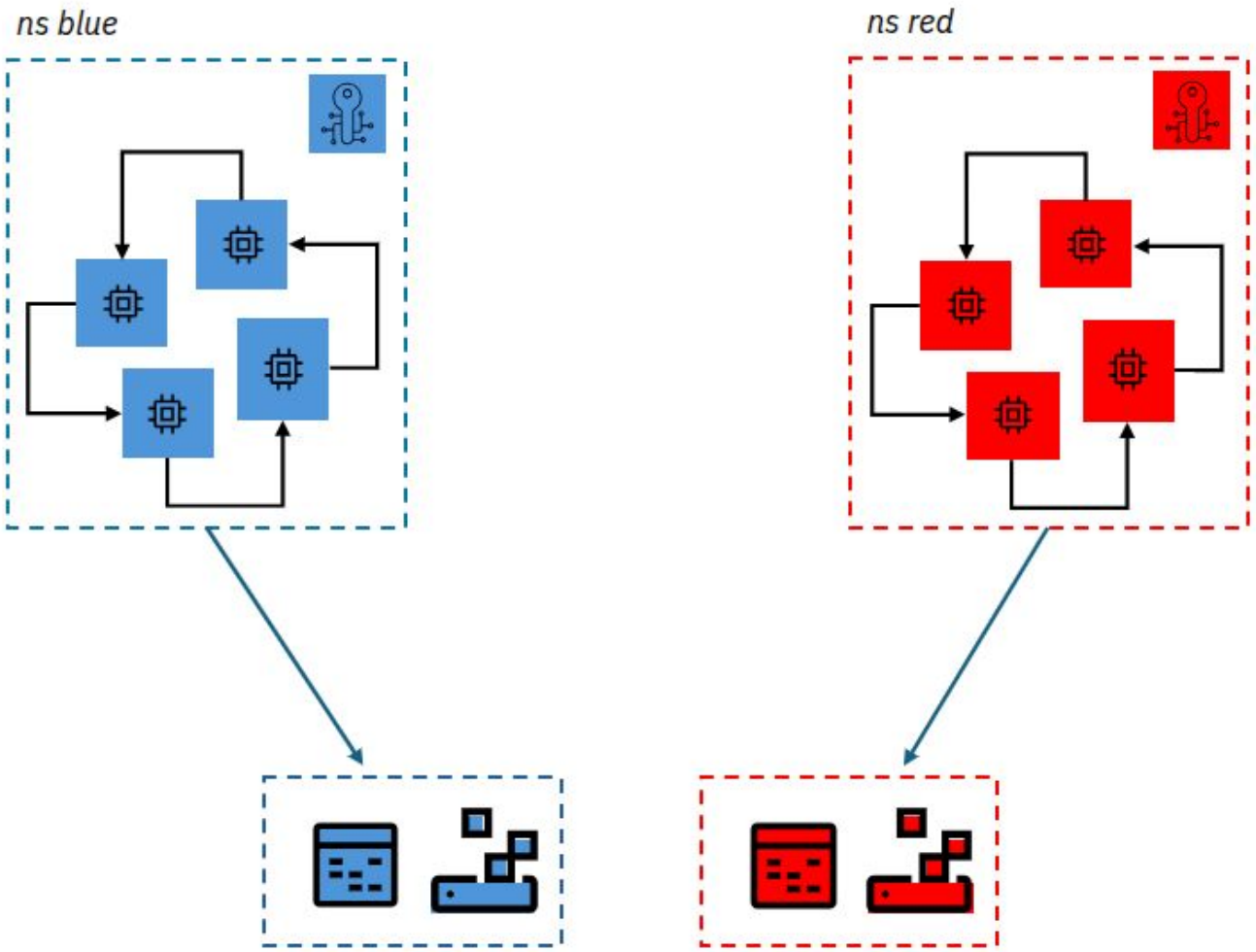
BigLake: BigQuery’s Evolution towards a Multi-Cloud Lakehouse

Core challenge for data Lakehouse Authorization

- Database-level access semantics , what users intuitively expect
- Storage-level enforcement , what zero-trust requires
- While maintaining direct paths, what performance and scalability demands

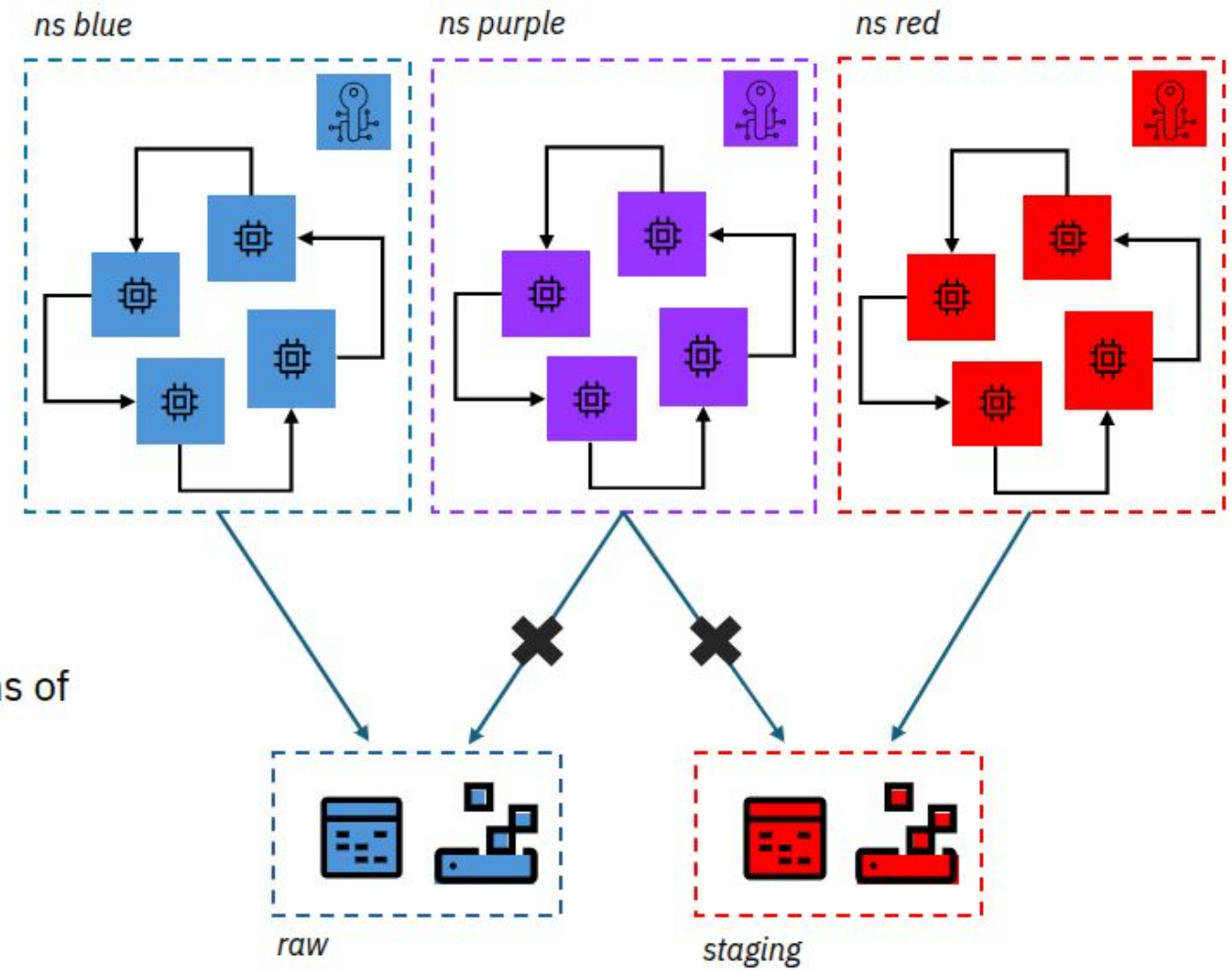
Access Control

Secure and govern: Namespace scoped secrets



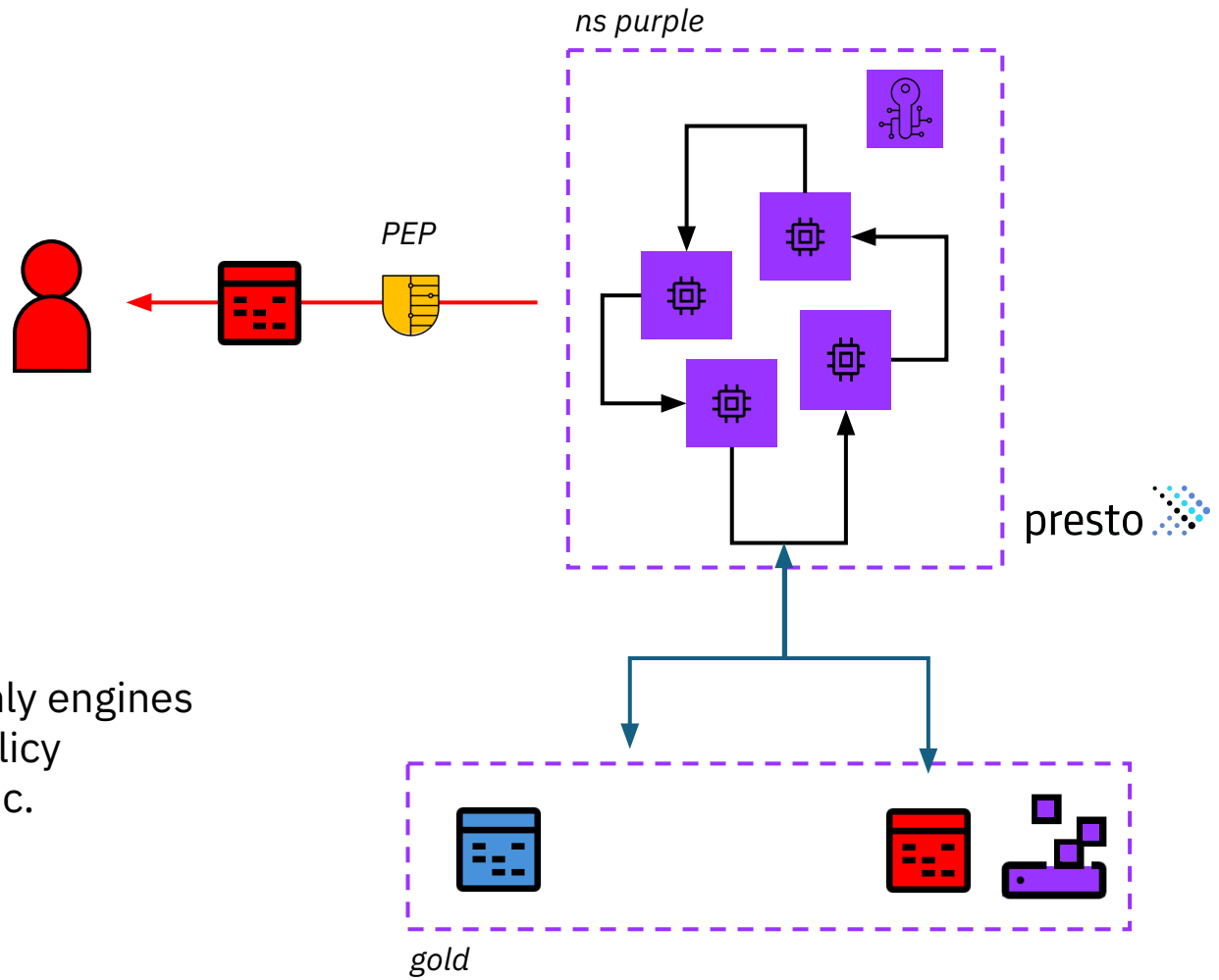
Buckets have 1:1 relationship with namespaces, dataset per bucket.

Secure and govern: Namespace scoped secrets



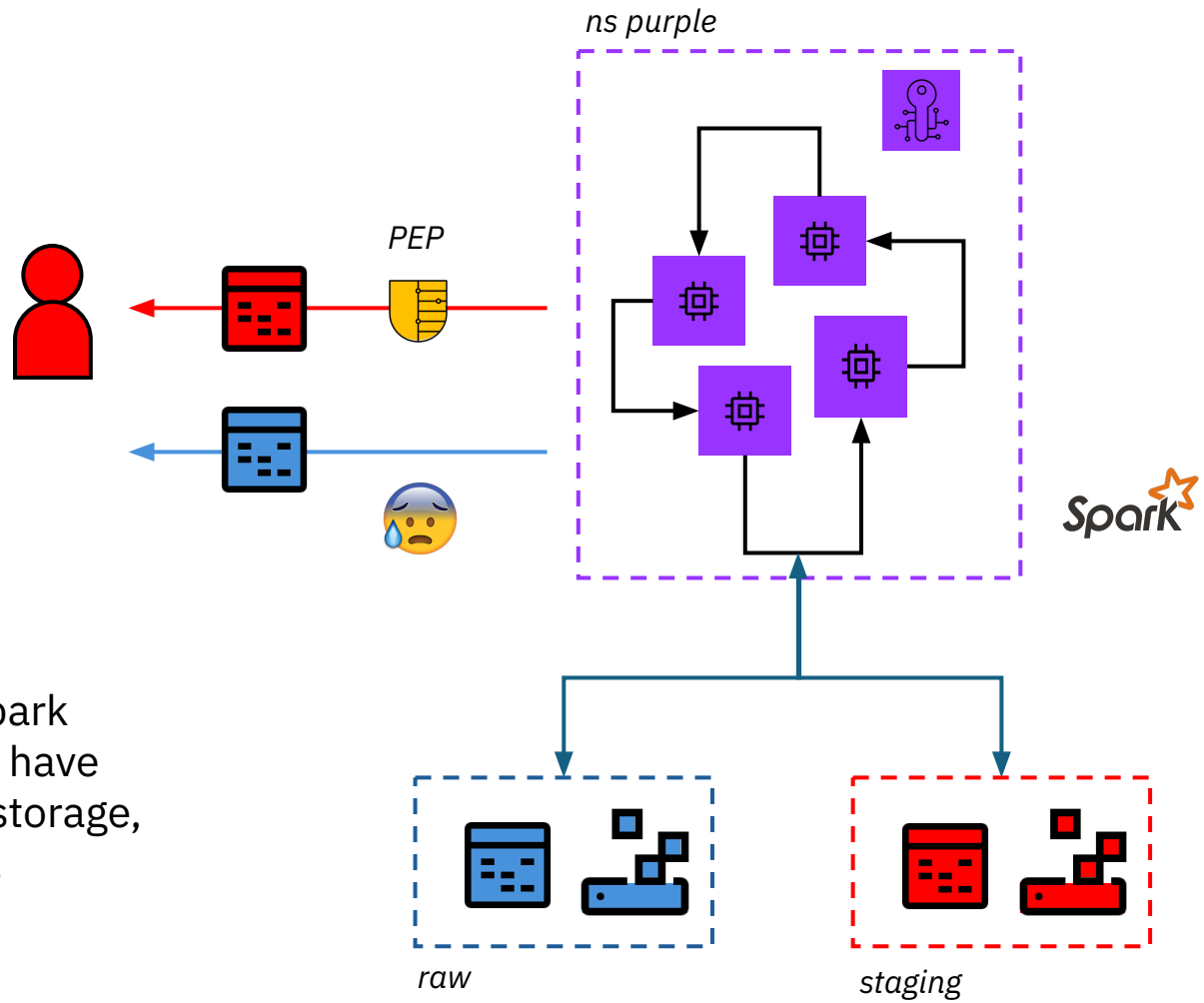
Highlights the limitations of ObjectBucketClaim for Lakehouse workflows.

Secure and govern: Engine policy enforcement



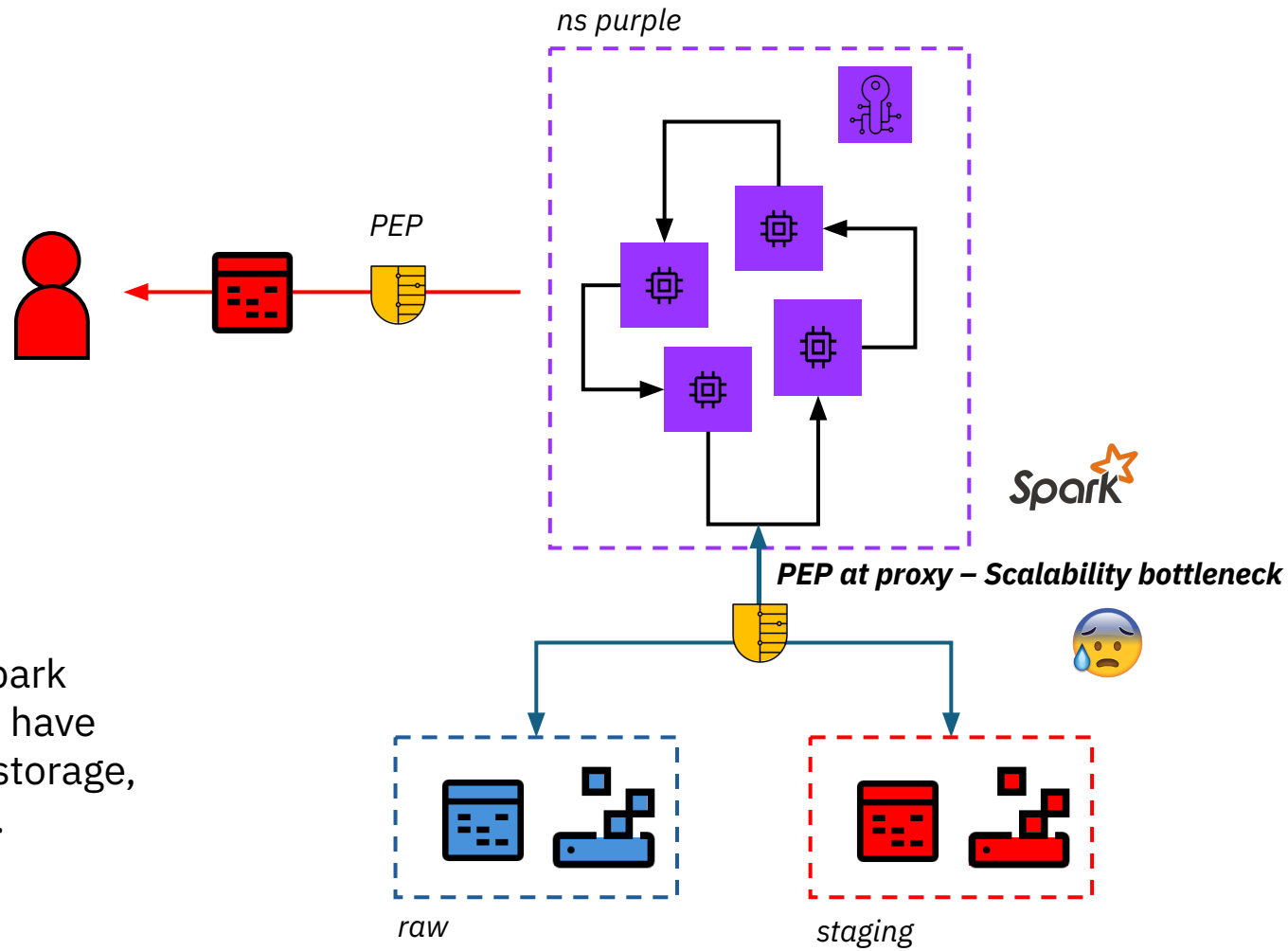
Viable for SQL only engines with common policy enforcement logic.

Secure and govern: Engine policy enforcement



Unsuitable for Spark applications that have direct access to storage, bypassing policy.

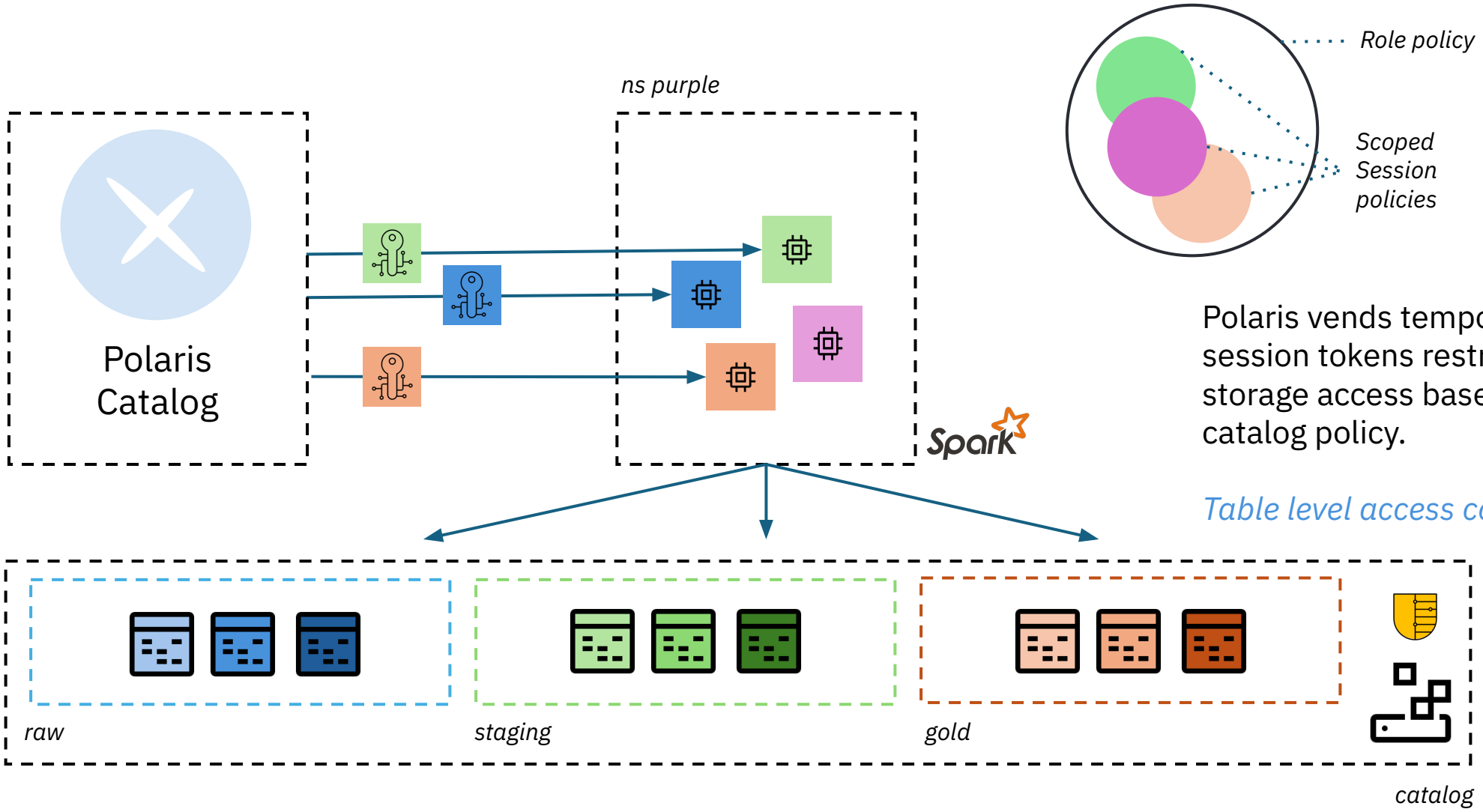
Secure and govern: Engine policy enforcement



Unsuitable for Spark applications that have direct access to storage, bypassing policy.

Technical Catalogs

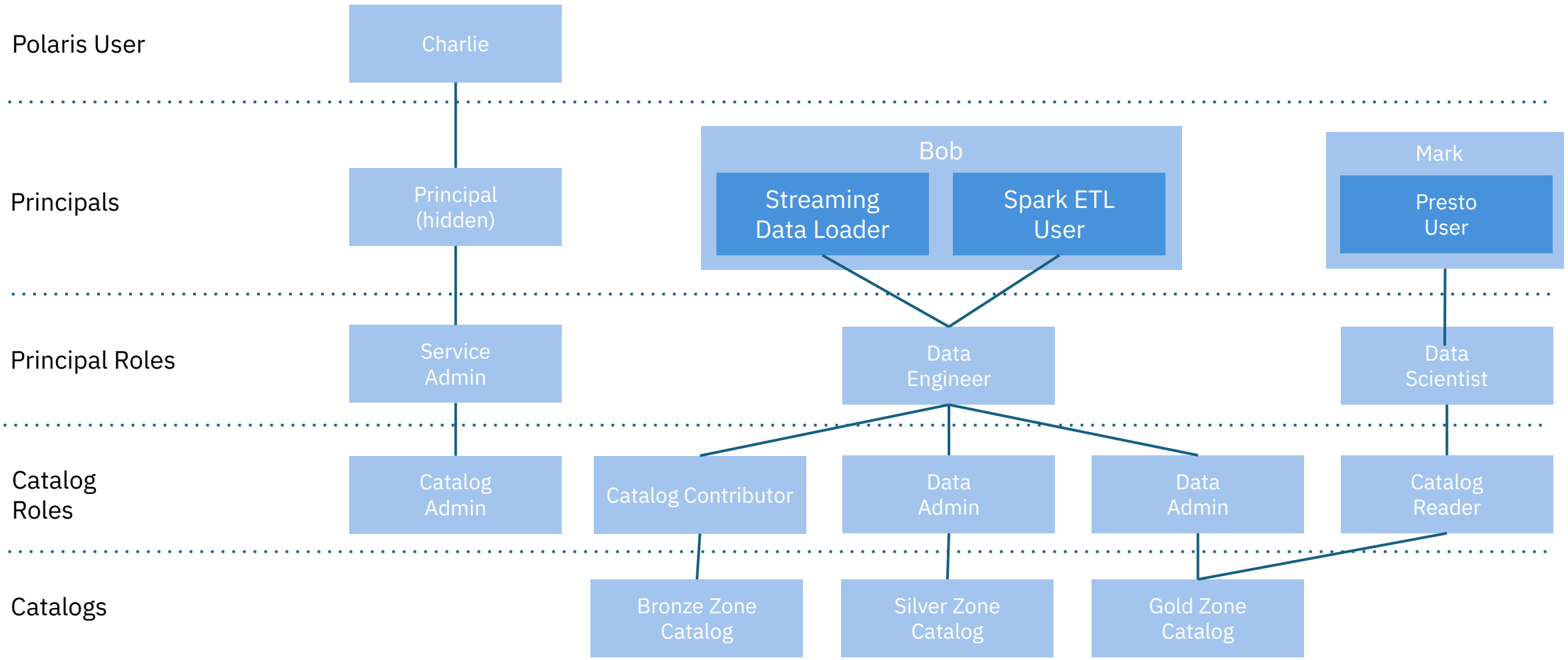
Secure and govern: Catalog credential vending



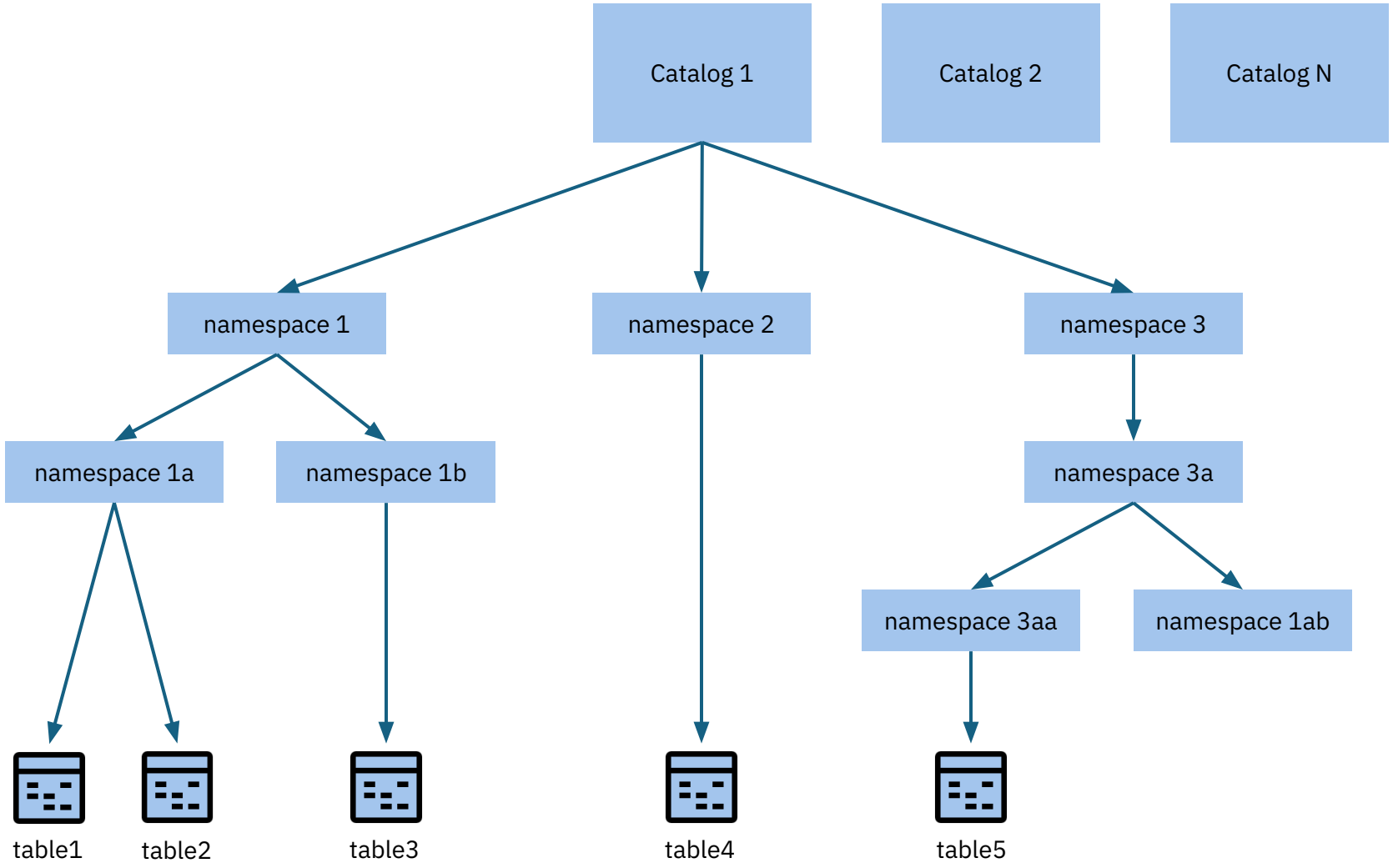
Polaris vends temporary session tokens restricting storage access based on catalog policy.

Table level access control

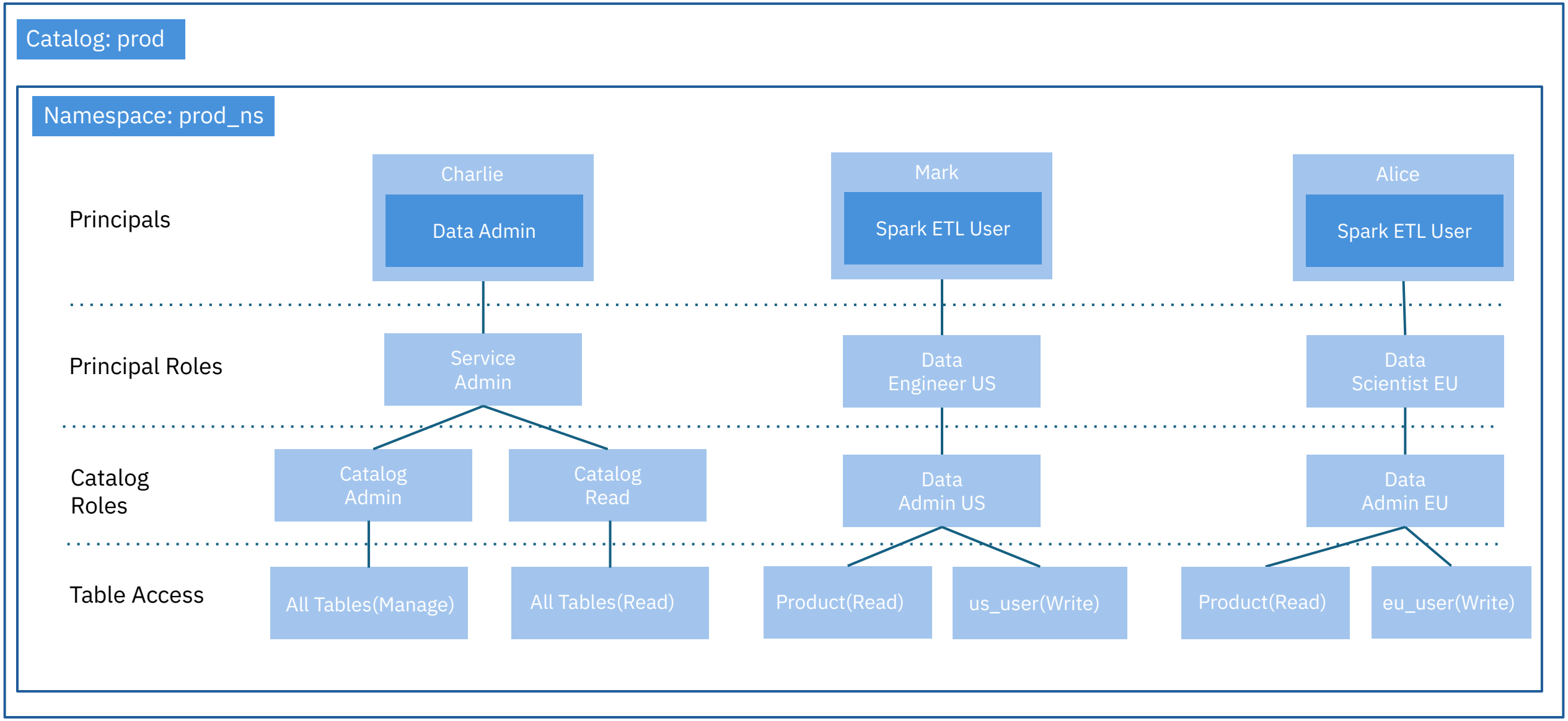
Secure and govern: RBAC example



Secure and govern: Namespaces



Secure and govern: RBAC granularity per table



Secure and govern: RBAC for other tables?

```
foo/                              # Iceberg table 'foo'
├── metadata/                     # Contains all table metadata files
│   ├── version-hint.text         # Points to current metadata version
│   └── v[version].metadata.json  # Metadata file for each version
├── data/                         # Contains data files
│   └── [partition_spec]/         # Optional partition directories
│   └── [file_group]/             # Groups of data files
│   └── [data_file].parquet       # Data files (usually Parquet)
└── snapshots/                    # Contains snapshot metadata
    └── snap-[id].avro            # Snapshot metadata files
```

```
s3://foo/                         # LanceDB root location
├── bar/                          # Table 'bar'
│   ├── data/                     # Column data files
│   ├── indices/                  # Index files
│   ├── _latest.manifest          # Points to current version
│   └── schema.arrow              # Table schema
└── baz/                          # Table 'baz'
    ├── data/
    ├── indices/
    ├── _latest.manifest
    └── schema.arrow
```

s3-tags are applied to these objects to map data files to tables and namespaces, and to use as conditions in session policies

Polaris like access controls could easily be extended to lance tables with a wrapper or native support for s3-tags and client side support for token vendors

Delta (and other format) Table Support in Polaris

Table optimization

Table compaction

- Combine small files into fewer, larger ones
- Merging delete files with data files generated by Iceberg merge-on-read
- Clustering tables by z-order
- Vector index regeneration

Snapshot Management

- Prune old table snapshots

Unreferenced File Removal

- Expire objects not referenced in any table snapshots

[Table Maintenance Integration with Polaris]

Terraform-Driven Ceph Object & Polaris Deployment

1

Automate Ceph RGW Prerequisites

- Provision S3 resources + IAM Roles and Policies

2

Deploy Polaris and Jupiter Notebook Services

- Spin-up Polaris Catalog Control Plane
- Launch Jupiter Notebook for Interactive Demo

3

Create and manage Polaris resources

- Create Principals, Roles & Grants
- Define Tables and schemas

4

Notebook: Catalog Credential Vending with Fine Grained Table RBAC

- Run SparkSQL queries using least privileged access

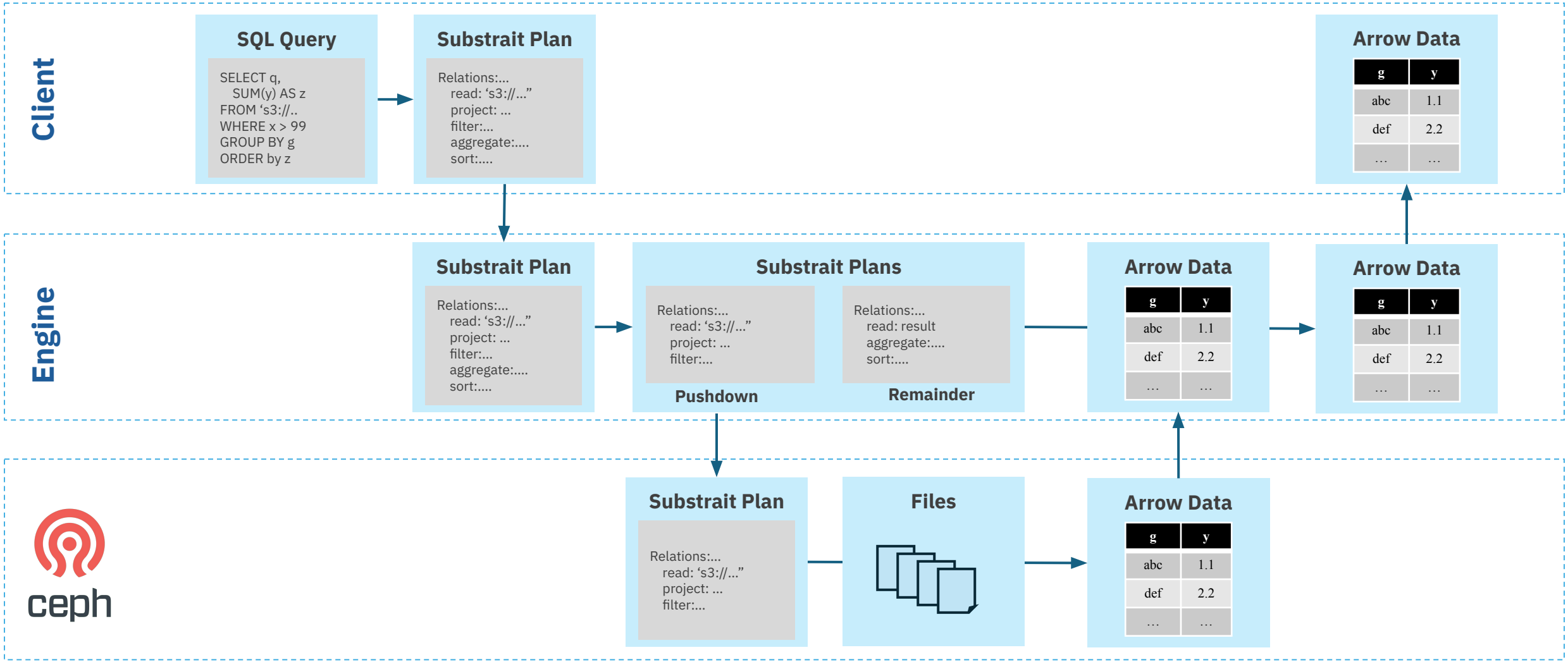
Questions ?

Extra

Read API for structured records

SQL

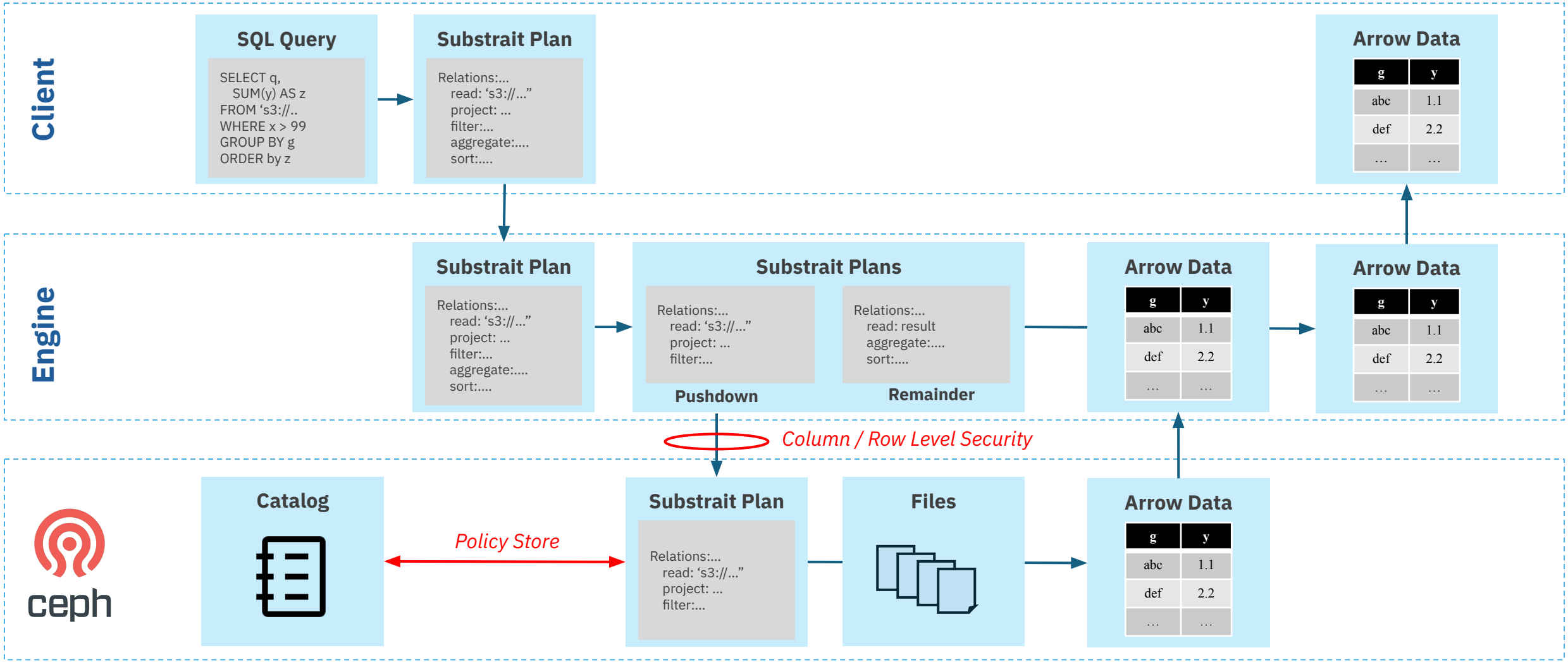
Volley
Read API



Read API for structured records: secure and govern

SQL

Volley
Read API



Directory tables

Add notion of Volumes and Directory tables to organize unstructured data

- Secure and govern unstructured like structured data
- Structured information about unstructured data
- AI use cases
  - Features (training), vectors (RAG), and full-text search through external indexes
  - FAISS or DiskANN for vector

- Bucket notifications to pub/sub for ISV software, serverless (Knative), AI extractors

Unstructured Data Support in Polaris