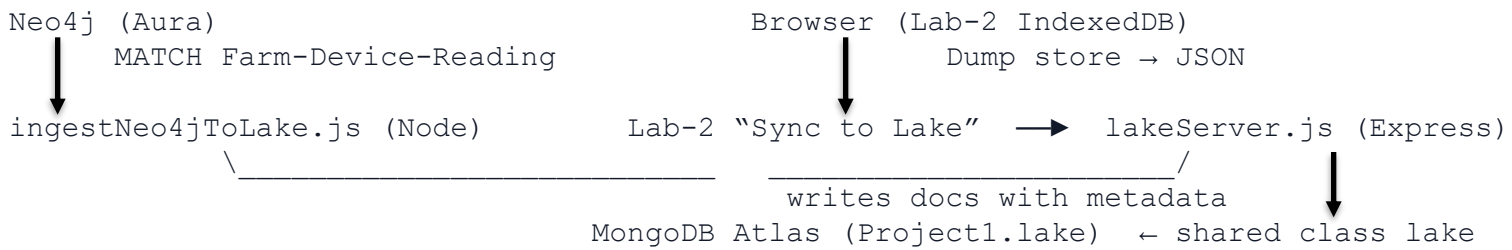
Architecture & Data Flow

IndexedDB + Neo4j → MongoDB Lake

- Sources: Neo4j (Aura) graph (Farm → Device → Reading) and Lab-2 Browser IndexedDB rows (id, reading/metric, timestamp, notes).
- Ingestion: ingestNeo4jToLake.js reads Cypher; lakeServer.js (Express) receives POST from Lab-2 and writes docs.
- Lake (shared Atlas cluster: lab2cluster.yub3wro): DB Project1, collection lake; each doc has: author, studentId, sourceDB, ingestedAt (UTC), tags.
- Validation: verified in MongoDB Compass using filters on author and sourceDB.



Lessons Learned

What I changed/verified while integrating the pipeline:

- Environment alignment: initially targeted wrong cluster; fixed .env to lab2cluster.yub3wro and restarted scripts.
- Author tagging in shared lake: added author & studentId to each doc; Compass filters become trivial.
- Neo4j driver: one statement per run; seeded with small MERGE batches for reliability.
- UTC timestamps: normalized to ISO for cross-system consistency.
- console.table output made Q1 verification clear.

Challenges & Next Steps

What worked well and what I would improve

- Worked well: simple Cypher join → clean JSON; minimal Express endpoint for Lab-2 syncing; filtering by author/sourceDB in Compass.
- Challenges: “bad auth” on Atlas when URI creds were placeholders; also queried the wrong cluster—fixed by reading runtime logs.
- Duplicate risk if re-running ingesters; next step: upserts (reading.rid for Neo4j, sensorId+timestamp for IndexedDB).
- Future: add indexes ({author:1}, {author:1, sourceDB:1}, {author:1, ingestedAt:-1}) and a tiny dashboard/aggregations (e.g., latest readings per farm).