Yes—here's a functioning **Layer 3 (warehouse orchestrator)** that "picks and chooses" datakits and dbt models per tenant, using a spec-first approach and mirroring prod behavior while staying dev-friendly.

📦 **Download:** layer3-warehouse.zip

What's inside

- Astronomer Runtime **3.0-10** project (Airflow 3.0.6+astro.1)
- **Spec-first** configs in `configs/warehouses/` (e.g., `acme.yaml`, `globex.yaml`) that declare:
  - Which **Bronze** tables to land (via your Layer-2 datakit image)
  - Which **dbt selects** to run for **Silver**, **Dims**, **Facts**
- **DAG factory** (`dags/warehouse_factory.py`) that reads the YAML spec and builds a full pipeline:
  - Bronze (per-table **DockerOperator** tasks)
  - dbt **Silver** → **Conformed Dims** → **Fact(s)**
- **Kerberos sidecar (Option B)** for local fidelity: sidecar refreshes TGT into a shared `krb_ccache` volume; tasks read `KRB5CCNAME=FILE:/krb5cc/krb5cc`
- Traefik labels and external `edge` network so the UI routes like prod: `https://airflow-dev.customer.localhost`

How it mirrors prod

- **Operators:** Locally we run DockerOperator with the **same runner images** you'll point KPO to in clusters.
- **Sidecar:** The Kerberos cache is a shared volume here; in K8s it's a pod volume—same contract.
- **Routing:** `.localhost` host shape matches `*.myco.com` so URLs and ergonomics transfer.

Quick start

1. Ensure your Traefik bundle + local registry + Postgres are already running as in your docs.
2. Push the runners (or reuse if present):

```
registry.localhost/etl/datakit-bronze:0.1.0
registry.localhost/analytics/dbt-runner:1.0.0
```

3. Unzip, then:

```
cd layer3-warehouse
cp include/.env.example .env     # tweak DB creds if needed
docker network create edge || true
astro dev start
```

4. Open https://airflow-dev.customer.localhost ↗.
   Trigger DAG `warehouse_build__acme_wh` (or switch to `globex.yaml` via `WAREHOUSE_SPEC`).

Verify it's doing the thing

- Bronze landed:

```
psql "postgresql://postgres:postgres@localhost:5432/pagila_wh" -c "\dt bronze.*"
```

- Silver / Dims / Facts present:

```
psql "...pagila_wh" -c "\dt pagila_silver.*"
psql "...pagila_wh" -c "\dt gold_mart.*"
psql "...pagila_wh" -c "\dt gold_rental.*"
```

Customizing tenants

- Create a new file under `configs/warehouses/` selecting the **bronze_jobs** tables and dbt selectors for that warehouse.
- Set `WAREHOUSE_SPEC` in `.env` to your new file; `astro dev restart`.