

SpecLens – Operational Governance & Versioning

1. Versioning Strategy

Code: Git with main/dev branches

Data: datasets_vN.csv

Models: model_vN.pkl

Each model is linked to:

- dataset version
- configuration hash
- training metrics

2. CI/CD & Automation

Training and CT scripts are automated.

Each retraining produces a new model version.

Rollback is possible by selecting previous artifacts.

3. Model Lifecycle

States: Draft → Trained → Evaluated → Deployed → Deprecated

Transitions are logged.

4. Monitoring & CT

Performance is periodically evaluated on new data.

If accuracy or recall falls below thresholds, ct_trigger.py:

- builds a new dataset
- retrains the model
- increments version

5. Incident Response

In case of anomaly:

- Switch to previous model
- Inspect drift metrics
- Trigger manual review