Phase 5 – Python Integration & BI Analytics Layer

Overview

Phase 5 extended the Airline Business Intelligence Database into a **fully integrated Python-based analytics environment**.
Where Phase 4 produced the SQL analytical layer (CTEs, window functions, recursive models), Phase 5 operationalized these insights using **Pandas, SQLAlchemy, Matplotlib, and Plotly**, enabling advanced EDA, BI storytelling, and visual analytics.

This phase focused on:

- Building a **clean database access layer** in Python
- Wrapping Phase 4 SQL into reusable query functions
- Conducting operational, network, commercial, and loyalty analysis
- Exporting **11 BI-ready visualizations** for final reporting
- Documenting insights and methodology in a reproducible notebook

All work for this phase was performed in:

- **notebooks/04_python_analytics.ipynb** — analytics, visualizations, insights
- **docs/phase_5_notes.md** — accompanying documentation
- **docs/phase_5_*.png** — exported visualizations

---

1. Python Analytics Environment

A standardized Python environment was created to ensure reproducibility and alignment with Phase 4's SQL logic.

Components Included

- SQLAlchemy PostgreSQL engine
- `.env` loader and safe credentials management
- Centralized data-access wrappers:
  - `get_engine()`
  - `get_df(sql, params=None)`
- Global Matplotlib theme:
  - Airline-BI style (white background, navy palette, bold titles, thicker axes)
- Optional Plotly support for interactive charts

Outcomes

The notebook now acts as a unified analytical workspace capable of:

- Pulling data directly from PostgreSQL
- Executing any Phase 4 SQL model
- Producing BI-ready figures with minimal code

This environment sets the foundation for Phase 6 reporting.

---

2. SQL-to-Python Analytics Layer

Phase 5 transformed the SQL models from Phase 4 into **production-quality Python functions**. These wrappers ensure that all business metrics can be queried consistently and reused in dashboards or reports.

Operational Functions

- Airline punctuality
- Delay distributions
- Monthly percent of flights delayed

Network & Route Functions

- Busiest airports
- Worst routes (delay × cancellation blend)
- Origin–destination flow extraction
- Airport coordinate mapping

Commercial Functions

- Monthly revenue trends
- Revenue by fare class
- Payment success by channel

Loyalty & Customer Functions

- Customer lifetime value (CLV)
- Top-value customers
- CLV cumulative distribution (Pareto analysis)

Design Principles

Each function is:

- Deterministic
- Fully SQL-backed

- Compatible with Pandas
- Ready for BI dashboards or automation

This forms the bridge between SQL analytics and business-facing tools.

---

3. Analytical Insights Produced

Phase 5 explored four key BI domains: **Operations, Network, Commercial, Loyalty**.

A. Operational Analytics

- Delay rates ranged from **68–84%** across months
- Clear reliability differences across airlines
- Delay distribution showed synthetic long-tail behavior common in real ops datasets

These metrics validate both BTS-derived data and synthetic flight behavior.

B. Network & Route Analysis

- Busiest airports identified small regional hubs due to synthetic routing
- Worst routes showed 100% cancellations or extreme delays (expected under synthetic randomness)
- A Plotly Sankey diagram visualized OD flows
- Latitude/longitude scatter validated spatial integrity of all airports

Despite synthetic variability, network modeling behaved exactly as expected.

C. Revenue & Commercial Insights

- Monthly revenue followed plausible seasonal patterns in 2025–2026
- Revenue mix remained stable:
  - Basic + Standard dominated volume
  - Business + First provided yield
- Payment success rates were uniformly low (14–15%) due to synthetic payment generator logic

These findings align closely with Phase 2’s synthetic revenue assumptions.

D. Loyalty & Customer Economics

- CLV distribution right-skewed (common for loyalty programs)
- Top 5% of customers contributed disproportionate value
- Cumulative CLV curve demonstrated Pareto-like concentration

The CLV models matched expectations from Phase 4’s window functions.

---

4. Visual Analytics Artifacts

A total of **11 BI-ready charts** were exported to `docs/`:

1. Monthly_Revenue_Trend.png
2. Monthly_Revenue_Trend_Interactive.png
3. Revenue_by_Fare_Class.png

4. Flights_Delayed_by_Month.png
5. Average_Arrival_Delay_by_Airline.png
6. Distribution_of_Flight_Delay.png
7. Payment_Success_Rate_by_Channel.png
8. Customer_Lifetime.png
9. Top_10_Customers.png
10. Airport_Map.png
11. Route_Sankey.png

These figures serve as the visual foundation for Phase 6 presentations and the final PDF report.

---

5. Notes on Future-Dated Records

During analysis, some data (e.g., flights, payments, bookings) included dates extending into **2025–2026**.
This behavior is intentional and originates from the **synthetic data generator** in Phase 2.

Key Reasons

- Synthetic generator assigns randomized schedule dates
- Airlines maintain future schedules months or years ahead
- Future-dated rows increase analytical volume for BI modeling
- They do *not* represent forecasting—simply enriched synthetic data

All metrics (revenue, CLV, delays) remain valid and meaningful.

---

6. Deliverables Produced in Phase 5

- `notebooks/04_python_analytics.ipynb`
- `docs/phase_5_notes.md`
- `docs/phase_5_*.png`
- Updated README (Phase 5 section)
- Updated CHANGELOG.md (Phase 5 entry)

These deliverables complete the Python analytics layer and prepare the project for its final phase.

---

7. Phase 5 Summary

Phase 5 successfully integrated Python into the BI workflow, producing:

- A robust SQL-to-Python analytics layer
- Full operational, network, commercial, and loyalty insights
- A rich suite of visual analytics
- Clean documentation for academic and portfolio use

Where Phase 4 delivered the analytical SQL engine,
Phase 5 transforms that engine into a full BI analytics platform, ready for dashboards, reports, and final presentation deliverables.