**NBA Ticketing & Renewal – Runbook (Ops & Handoff)**

**0) Architecture (mental map)**

* **Data source:** synthetic CSVs → stg\_→ Dim/Fact (Azure SQL)
* **Orchestration:** ADF pipeline PL\_Master\_Daily → runs dbo.sp\_Run\_Daily\_Load
* **Forecasting:** forecast\_job.py writes to dbo.FactForecast (aggregate + per-game)
* **Semantic layer:** Power BI (Import) with Incremental Refresh + RLS (UPN→TerritoryId)

**1) Daily operations (what actually runs)**

**03:00 AM** – ADF trigger → PL\_Master\_Daily

* Executes all sp\_Load\_\* in order (dims → facts).
* Each proc is idempotent (upsert/merge).

**03:15 AM** – (Optional) Forecast job

* Run python forecast\_job.py on your scheduler (Automation/VM/etc.).
* Clears future rows, writes 56 days of **Attendance + Revenue** (GameId=0) and per-game forecasts.

**Power BI Service** (after publish)

* Dataset refresh per your schedule (Incremental Refresh active).

**Daily quick check (5 min)**

-- 1) Yesterday’s loads changed rows?

SELECT 'TicketSales', COUNT(\*) FROM dbo.FactTicketSales WHERE CAST(ModifiedAt AS date) = CAST(SYSDATETIME() AS date)

UNION ALL SELECT 'Attendance', COUNT(\*) FROM dbo.FactAttendance WHERE CAST(ModifiedAt AS date) = CAST(SYSDATETIME() AS date)

UNION ALL SELECT 'Renewals', COUNT(\*) FROM dbo.FactRenewals WHERE CAST(ModifiedAt AS date) = CAST(SYSDATETIME() AS date);

-- 2) Forecast present for next 7 days?

SELECT Metric, COUNT(\*) FROM dbo.FactForecast WHERE ForecastDate BETWEEN CAST(SYSDATETIME() AS date) AND DATEADD(day,7, CAST(SYSDATETIME() AS date)) GROUP BY Metric;

**2) One-time configuration (recap)**

* **SQL:** all scripts executed; DimDate populated; view: dbo.vw\_FactAttendance (includes GameDate).
* **ADF:** Linked Service LS\_AzureSqlDB, pipeline PL\_Master\_Daily with schedule.
* **Power BI Desktop:**
  + Parameters: RangeStart/RangeEnd (Date/Time)
  + Filters (folding):
    - FactTicketSales[SaleDate] between RangeStart/RangeEnd
    - vw\_FactAttendance[GameDate] between RangeStart/RangeEnd
  + Incremental Refresh: store 5y, refresh 60d, detect changes = ModifiedAt
  + Relationships:
    - DimDate[Date] (1) → FactTicketSales[SaleDate] (\*)
    - DimDate[Date] (1) → vw\_FactAttendance[GameDate] (\*)
    - DimDate[Date] (1) → FactForecast[ForecastDate] (\*)
    - DimGame[GameId] (1) → FactTicketSales[GameId]/vw\_FactAttendance[GameId]/FactForecast[GameId] (\*)
  + RLS role TerritoryUser (UPN→TerritoryId mapping via SecurityUserMap).

**3) Manual procedures (common tasks)**

**Run the full load now**

* **ADF Studio → PL\_Master\_Daily → Debug**
* Or in SQL: EXEC dbo.sp\_Run\_Daily\_Load;

**Run the forecast now**

cd <repo>\python

python forecast\_job.py

**Clear & re-seed forecasts (safe re-run)**

DELETE dbo.FactForecast WHERE ForecastDate >= CAST(SYSDATETIME() AS date);

**Publish/Refresh Power BI**

* Desktop → Publish to workspace.
* Service → Dataset → Refresh now (incremental kicks in service).

**4) Monitoring & alerts**

**Azure Data Factory**

* **Monitor** blade: pipeline run status, activity outputs.
* Enable **Diagnostic settings** → Log Analytics; create alert:
  + Condition: Failed pipeline runs > 0 (last 1h) → Action: email/Teams webhook.

**SQL (basic health)**

-- Late/no forecast signal (should always be >0)

SELECT COUNT(\*) FROM dbo.FactForecast WHERE ForecastDate = CAST(SYSDATETIME() AS date);

-- Duplicates in stg tickets (should be 0)

SELECT TOP 1 TicketId, COUNT(\*) c FROM dbo.stg\_TicketSales GROUP BY TicketId HAVING COUNT(\*)>1;

**Power BI Service**

* Dataset → **Refresh history**.
* Configure failure alerts to owners.

**5) Troubleshooting (fast fixes)**

**A. Power BI refresh fails with folding/IR errors**

* Make sure the attendance model uses **dbo.vw\_FactAttendance**, not the base table.
* Confirm RangeStart/RangeEnd filters are applied **directly** on SaleDate/GameDate steps (no merge before the filter).
* If detect-changes set, ensure column ModifiedAt exists in the query (keep it in the view).

**B. “Duplicate value on one side of relationship” (FactForecast)**

* Ensure **DimDate** is on the **one** side, **FactForecast** on the **many**.
* Delete/recreate the two relationships exactly as noted above.

**C. ADF pipeline fails a fact MERGE (duplicates)**

* Check duplicate keys in staging (e.g., stg\_TicketSales duplicate TicketId).
* Run dedupe:

;WITH x AS (

SELECT \*, ROW\_NUMBER() OVER(PARTITION BY TicketId ORDER BY ISNULL(SaleDate,'1900-01-01') DESC, ISNULL(OrderId,0) DESC) rn

FROM dbo.stg\_TicketSales

) DELETE FROM x WHERE rn>1;

* Re-run: EXEC dbo.sp\_Load\_FactTicketSales;

**D. Forecast job errors**

* **ODBC driver missing** → install ODBC 17/18 for SQL Server.
* **Prophet not installed** → script falls back to MA28 for attendance; otherwise pip install prophet==1.1.5.
* **No revenue forecasts** → with our patch, MA28 is always produced.

**6) Performance tuning (when volume grows)**

* **SQL indexes**
  + FactTicketSales(SaleDate) + ModifiedAt
  + FactAttendance(GameId) + ModifiedAt
  + FactForecast(GameId, ForecastDate, Metric) (already recommended)
* **Columnstore (optional for big data):**

CREATE CLUSTERED COLUMNSTORE INDEX CCI\_FactTicketSales ON dbo.FactTicketSales;

* **Power BI**
  + Keep **staging** queries “Enable load” **off**.
  + Consider Aggregations or Composite models if you exceed Import limits.

**7) Security & access**

* **RLS:** enforced in Power BI via SecurityUserMap (UPN→TerritoryId); no rep-level exceptions.
* **Secrets:** .env for dev; move to **Key Vault** (linked service or Automation variables) for prod.
* **SQL firewall:** restrict to required IPs; “Allow Azure services” ON for ADF/Service.

**8) Change management**

* **SQL changes** via numbered scripts; keep a /sql/patches folder.
* **ADF**: export ARM template after changes.
* **Power BI**: version PBIX files in source control (even if binary).
* **Docs:** update /docs/data\_dictionary.md when columns/tables change.

**9) Backup/restore**

* **SQL**: Export **BACPAC** (Azure Portal) before major changes.
* **ADF**: Export ARM template (Manage → ARM).
* **PBIX**: keep latest copy offline (OneDrive/SharePoint).

**10) Go-live checklist**

* ADF PL\_Master\_Daily trigger shows **Scheduled**.
* Forecast job runs once nightly (Automation or VM).
* Power BI dataset refresh scheduled; Incremental Refresh policy verified in Service.
* RLS tested with two real UPNs.
* Monitor/alerts wired (ADF + Power BI).