SRT Desk project task

Notes
Please don't spend much more than 60 mins or so on the task. This is more about how you attack the task, how you break it down into discrete parts, and choose where to do the actions. Put your thoughts and any relevant small code fragments into a Powerpoint deck. No need to make it look particularly nice, just enough that it can be read and understood!!

System-Design Question — Dual-Source Ingestion, Transformation & Aggregation in Excel

Scenario
• **Source A (Database):** A SQL view dbo.Trades (~2 million rows) with TradeID, Book, Counterparty, TradeDate, Notional, Currency, IR_PV.
• **Source B (Flat-File):** A daily FX-rate CSV, FxRates_YYYYMMDD.csv, stored in \\GlobalShare\FX\. Fields: Ccy, SpotRateToUSD. Exactly one file per day; size ≤ 1 MB.

Business Need
Build an Excel-driven workflow that, on demand, produces an aggregated table of IR_PV_USD = IR_PV / SpotRateToUSD, grouped by **Book** and **Counterparty**, for a user-specified date range (StartDate, EndDate entered on a control sheet).

Key expectations:
1. Refresh time < 30 seconds on a standard trader workstation
2. Minimal workbook bloat (avoid loading all raw rows into visible sheets).
3. Robustness to:
   • Missing or malformed FX file
   • Schema drift (extra / reordered columns)
   • Database latency or temporary outage
4. Clear audit trail of when data were refreshed and by whom.
5. Solution must survive Excel version upgrades and be supportable by the ops team.

Open-Ended Task
Design and outline an **optimised end-to-end solution** that meets the requirements above.

You are free to choose and combine any Excel-friendly technologies (e.g., Power Query, VBA/ADO, Office Scripts, Data Model, external ODBC links, Python add-ins, etc.).

Your deliverable should cover:
• Architecture diagram or narrative: data flow from each source into Excel, transformation steps, join logic, and final aggregation.
• How you will parameterise the date filter so the database query is "folded"/pushed down (or equivalent optimisation) rather than post-filtered in Excel.

• Strategy for identifying and loading the **most-recent** FX CSV without hard-coding the filename.
• Memory/performance optimisations (e.g., connection-only queries, staged loads, background refresh, batching, streaming, multi-threading, compression).
• Error-handling and logging approach.
• Incremental-refresh or delta-load design to avoid re-pulling the entire 2 million-row table daily.
• Governance: version control of queries/macros, change-management, and security of database credentials.

Present your answer as clearly structured **short** slide deck; include code fragments **only** where needed to demonstrate critical details.