CASE STUDY

Automated Data Pipeline for a Proptech Company

A company in the proptech space needed to aggregate, grade, and enrich millions of professional records from public listing APIs. A process that took two weeks of manual Excel work now runs in six hours — unattended.

SECTOR	DURATION	STACK	STATUS
Proptech / Real Estate Tech	**4 Weeks**	**Apache Beam / GCP Dataflow**	**In Production (Quarterly)**

The Challenge

The client's sales team depended on a graded list of industry professionals across the United States — ranked by transaction volume, listing history, and activity level. This data came from multiple public API sources and needed to be aggregated per individual, scored against configurable thresholds, and exported as enriched CSVs for their outreach tools.

The existing process was entirely manual: a team member would spend roughly **two weeks per cycle** pulling data through spreadsheets, copy-pasting between tabs, and manually calculating grades. The process ran quarterly, and each cycle was error-prone and exhausting.

What We Built

PHASE 1 — PROTOTYPE

We started with a custom Python application that dynamically paginated through the API, aggregating records per individual until configurable thresholds were met (e.g., $40M total volume for the top tier). This prototype validated the data model and API integration patterns, handling rate limiting, retries with exponential backoff, and malformed response recovery.

PHASE 2 — PRODUCTION PIPELINE

The production system was rebuilt on **Apache Beam**, deployed as a **Google Cloud Dataflow** template. This gave the client a managed, serverless pipeline that auto-scales workers based on data volume — no infrastructure to maintain.

Pipeline Architecture: 3-stage fan-out design. Stage 1 fetches profile data and calculates initial scores. Stages 2 and 3 paginate through active and historical records respectively, with iterative loop unrolling (up to 19 pages deep) and consecutive-failure circuit breakers. All stages write raw API responses to BigQuery for audit and reprocessing.

PHASE 3 — SELF-SERVICE DASHBOARD

A Flask-based web dashboard lets the client's team trigger pipeline runs with configurable parameters: grade thresholds, date cutoffs, whether to include active records, and which tiers to process. The dashboard writes batch tracking records to BigQuery and monitors pipeline status — the team runs it themselves without engineering support.

Results

93%

Reduction in processing time (2 weeks → 6 hours)

~20/s

Sustained API calls per second with rate limiting and retries

15 min

Human time per cycle (configure and trigger)

Technical Highlights

- **Serverless scaling** — Dataflow auto-provisions workers based on data volume; no servers to manage
- **Configurable grading** — Thresholds for each tier (A+ through F) set per run via dashboard
- **Smart pagination** — Early termination for top-tier individuals (skip remaining pages once threshold met)
- **Resilient API handling** — Exponential backoff, consecutive-failure circuit breakers, malformed response recovery
- **Full audit trail** — Every API response stored in BigQuery with batch ID for reproducibility
- **Enrichment pipeline** — Missing contact details filled via secondary enrichment tools post-export

Apache Beam · Google Cloud Dataflow · BigQuery · Cloud Storage · Secret Manager · Python · Flask · REST APIs

Client Impact

The pipeline runs quarterly with zero engineering involvement. The client configures parameters through the dashboard, triggers the run, and receives graded export files in Cloud Storage. What used to consume a team member for two full weeks is now a 15-minute task followed by an automated 6-hour pipeline execution.

The system has been running in production for multiple cycles with consistent results. The client has expressed strong satisfaction with both the reliability and the self-service nature of the solution.

GTA Labs | Multilevel Performance Services Inc. greg@gtalabs.com | gtalabs.com | 416.556.9690