Data Correction Strategies for Policy Ingestion

1. Overview

When ingested policy data contains errors (e.g., incorrect phone numbers, names, or addresses), two approaches enable corrections without full re-ingestion:

- **Delta/Correction File** – Bulk corrections via file upload
- **Correction/Patch API** – Single-record corrections via REST API

2. Strategy 1: Delta/Correction File

> **Definition**
>
> A bulk correction mechanism wherein a CSV containing only the corrected rows is uploaded through the existing Ingestion pipeline. The Processing Service detects it as a correction (via metadata or parameter) and performs UPDATE operations instead of INSERT, using Policy Number and PAN as unique identifiers.

When to Use

- 50+ records requiring correction in a batch
- Same type of error across multiple records (e.g., phone format)
- Corrected data is available in Excel/CSV format
- Reuse of the existing upload pipeline is preferred

Process Flow

1. **Create correction file** – CSV/Excel with key columns (Policy Number, PAN) for matching and corrected columns (e.g., Mobile, Customer Name). Include only rows that need fixing.
2. **Upload** – Use the standard ingestion upload flow with a correction indicator (e.g., `fileType=correction` query parameter or filename convention `*_correction.csv`).
3. **Ingestion Service** – Validates, stores the file, and creates a job (same as regular upload).

4. **Processing Service** – In correction mode:
    - Looks up each row by Policy Number and/or PAN
    - If found → UPDATE existing customer/policy with new values
    - If not found → Log warning, skip (do not create new records)
5. **Job lifecycle** – UPLOADED → PROCESSING → COMPLETED (or FAILED).

Benefits

- Uses existing ingestion pipeline; no new screens or flows
- Only modified records are processed; no full re-upload
- Audit trail via ingestion job ID, upload timestamp, and user
- Avoids duplicates when implemented as update-only
- Can be retried or re-run if needed

Considerations

- Requires a clear way to mark a file as a correction
- Processing Service must support both INSERT (normal) and UPDATE (correction) paths
- Unique identifiers (Policy Number, PAN) must be reliable

---

3. Strategy 2: Correction/Patch API

Definition

Explicit REST endpoints for updating individual customer and policy records. Used by support staff or admins via an internal correction tool or admin portal. Corrected values are submitted per record; services apply the update and log the change for audit compliance.

When to Use

- 1–5 records requiring correction
- Immediate correction is needed
- Corrections originate from support calls or admin review
- Granular audit trail (field-level) is required

Process Flow

1. **Locate record** – Support/admin finds the customer or policy by Policy Number, PAN, or Customer ID.
2. **Submit correction** – User enters only the corrected fields in a form or tool.
3. **API call** – Portal invokes `PATCH /api/v1/customers/{customerId}` or `PATCH /api/v1/policies/{policyId}` with the payload.
4. **Service updates** – Customer/Policy Service validates, applies the update, and records who changed what and when.
5. **Result** – Change is reflected immediately; user can verify on next refresh.

Benefits

- Precise, record-level correction
- Immediate effect without batch processing
- Audit trail: updatedBy, updatedAt, reason
- No file upload or reprocessing overhead
- Works for ad-hoc corrections (mobile, name, address, etc.)

Considerations

- Requires new endpoints and a correction UI/tool
- Access control and role-based permissions must be enforced
- May require mandatory "reason for change" for compliance

Example API Payloads

Update customer mobile number:

```
PATCH /api/v1/customers/cust_7x9k2
Content-Type: application/json

{
  "mobile": "9988776655",
  "reason": "Customer verified change during support call"
}
```

Update policy customer name:

```
PATCH /api/v1/policies/pol_3f8k1
Content-Type: application/json

{
```

```
  "customerName": "Amit Ramesh Kulkarni",
  "reason": "Middle name addition – document verified"
}
```

---

4. Strategy Selection Matrix

Factor	Delta/Correction File	Correction/Patch API
Volume	Bulk (10+ records)	Few records (1–5)
Source of correction	File (CSV/Excel)	Manual entry via UI
Typical user	Data ops, back office	Support agent, admin
Urgency	Batch; can be scheduled	Immediate
Implementation effort	Processing Service logic extension	New endpoints + UI
Audit	Job-level (job ID, upload metadata)	Field-level (who, when, reason)

---

5. Recommendation

- **Delta/Correction File** – For systematic, bulk data corrections originating from data teams.
- **Correction/Patch API** – For ad-hoc, single-record corrections by support or admin users.

Both strategies can coexist and complement each other within the same system.
