Diabetes 30-Day Readmission Project

Phase 1: Bronze Layer Implementation

```
STATUS       :: COMPLETE
ENVIRONMENT  :: DATABRICKS / PYSPARK
OBJECTIVE    :: RAW DATA INGESTION & LINEAGE PRESERVATION
ARTIFACT_ID  :: DIABETES_READMISSION_BRONZE
```

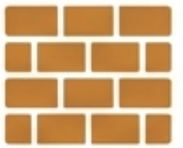
The Bronze Philosophy: Immutable & Raw

Preserve Lineage

The Bronze layer stores the raw dataset exactly as received from the source. No transformations are applied, ensuring a perfect audit trail against original source files.

Enable Reproducibility

By persisting the raw state, downstream logic changes can be re-run safely from this layer without needing to re-ingest external files or access source systems again.



Isolate Logic

Decoupling ingestion from transformation ensures that simple file transfer failures are isolated from complex business logic errors in later stages.

Source: Architecture Implementation Note, Notebook Cell 1.

Ingestion Logic: Constructing the DataFrame

```
raw_df = (
  spark.read
    .option('header', 'true')
    .option('inferSchema', 'true')
    .csv('/Volumes/diabetes_readmission/healthcare/diabetic_data.csv')
)
```

Header Parsing

Critical for correctly interpreting the first row as metadata (encounter_id, patient_nbr) rather than data values.

Auto-Typing

Allows Spark to detect data types (Integer vs String) during the initial pass, reducing manual DDL overhead.

Volume Mount

Data sourced from the 'healthcare' volume. Verified file size: ~19.1 MB.

Persistence Strategy: The Delta Write

```
bronze_df.write.format('delta') \\
  .mode('overwrite') \\
  .saveAsTable('diabetes_readmissions.bro
    nze_diabetes_readmission')
```

Configuration Explanations

Format: Delta

Utilizes the open-source Delta Lake protocol for ACID transactions and scalable metadata handling.

Mode: Overwrite

Ensures idempotency. Re-running the notebook replaces the table entirely, preventing duplicate data ingestion.

saveAsTable

Registers the data in the Hive metastore, making it immediately queryable via SQL.

Verification: Volume Integrity

```sql
SELECT COUNT(*)
FROM diabetes_readmissions.bronze_diabetes_readmission;
```

101,766

TOTAL ROWS INGESTED

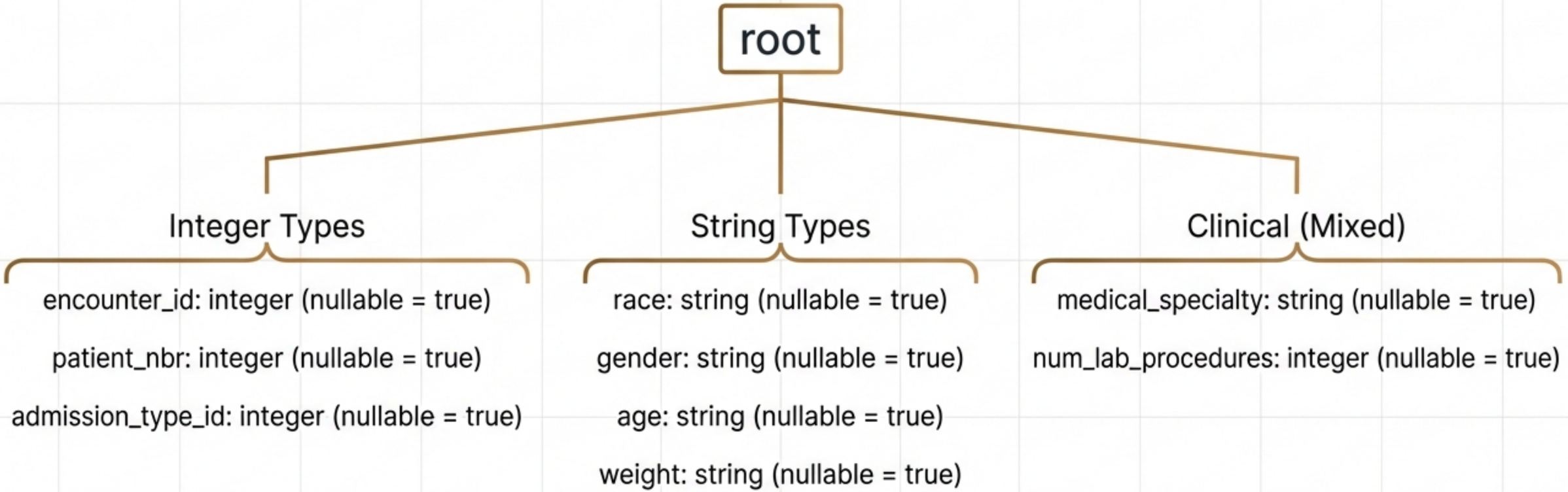
Validation Successful: All patient encounter records from the source CSV were parsed and written to the Delta table without data loss.

Data Profile: Attribute Preview

encounter_id id	**patient_nbr** patient_nbr	**race** Inter	**gender** typeface	**age** typeface	**weight** value	**admission_type_id** Inter typeface
2278392	8222157	Caucasian	Female	[0-10]	?	6
149190	55629189	Caucasian	Female	[10-20]	?	1
64410	86047875	AfricanAmer…	Female	[20-30]	?	1
500364	82442376	Caucasian	Male	[30-40]	?	1
16680	42519267	Caucasian	Male	[40-50]	?	1

Data Quality Flag: Missing values are represented by '?' rather than NULL. This requires cleaning in the Silver Layer transformation.

Schema Definition & Type Inference



Status: Schema inference successful. Mixed data types (Int/String) correctly identified without manual DDL.

Phase 1 Status: Ready for Transformation

Completion Checklist

- ✓ Raw CSV ingestion pipeline configured.
- ✓ Delta Lake table registered (`bronze_diabetes_readmission`).
- ✓ Volume validation (101,766 records confirmed).
- ✓ Initial schema inference validated.

Next Steps: The Silver Layer

1. Cleaning: Convert '?' placeholders in weight column to NULL.
2. Standardization: Normalize column names and cast integers to categorical types.
3. Enrichment: Join with mapping tables for `admission_type_id` decoding.

→

Proceed to Silver