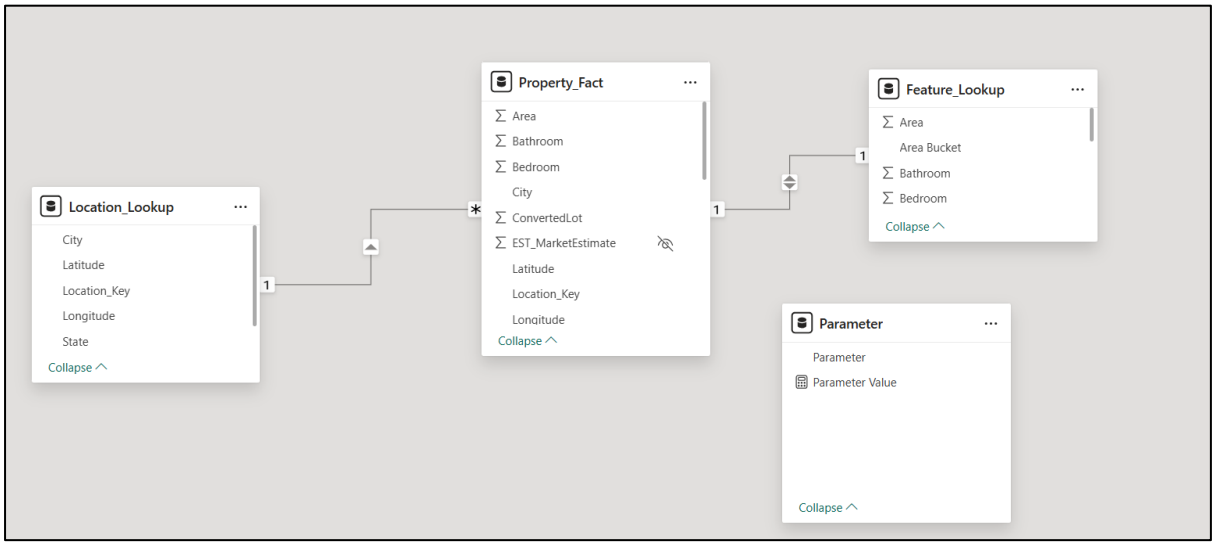
Technical Document



Relationship Summary:

From Table	To Table	Cardinality	Reason
Location_Lookup	Property_Fact	1 → *	One location, many properties
Feature_Lookup	Property_Fact	1 → 1	One Property, One Feature
Parameter	None	Disconnected	Scenario simulation via DAX

Power Query Data Cleaning & Transformation Steps

(Property Listings Dataset)

1 Data Ingestion

- Imported property listing data from a **CSV file**
- Explicit delimiter, encoding, and column count defined to avoid schema drift

---

2 Header & Data Type Standardization

- Promoted first row as column headers
- Initially cast all columns as **text** to prevent early conversion errors
- Later converted relevant fields to **numeric types**:
    - Bedroom, Bathroom

- Area, Price, PPSq
- LotArea, ConvertedLot
- MarketEstimate, RentEstimate

Why: Safe two-step typing avoids broken refresh due to malformed values.

3️⃣ Invalid & Missing Value Handling

- Removed rows where State = "nan"
- Replaced **conversion errors** with null
- Filled missing values with **business-default constants**, e.g.:
  - Bedroom → 3
  - Bathroom → 2
  - Area → 1840
  - PPSq → Median-derived value
  - MarketEstimate → recalculated later
  - RentEstimate → capped default

Why: Ensures continuity in aggregations and scenario analysis.

4️⃣ Text Cleansing (Street Address)

- Trimmed whitespace
- Removed special characters
- Converted to **Proper Case**
- Constructed a fallback address using City + Zipcode if street was missing

Result: A standardized, user-friendly **Street Address** column.

5️⃣ Duplicate & Noise Reduction

- Removed duplicate records using a composite of:
  - Street Address
  - Zipcode
  - Bedroom
  - Area
  - Price

Why: Prevents double-counting properties in market value calculations.

---

6 Outlier & Data Quality Filters

Applied multiple **business-rule filters**, including:

- Area between **150 and 10,000 sq ft**
- Price between **20,000 and 5,000,000**
- RentEstimate between **500 and 10,000**
- Lot Size between **500 and 500,000**
- Bedroom values limited to **0–5**
- LotArea < 11 (unit sanity check)

Why: Removes unrealistic properties that would skew analytics.

---

7 Unit Normalization (Lot Size)

- Converted lot size into a **single unit (sq ft)**:
    - Acres → multiplied by 43,560
    - Sqft → retained as-is

Why: Enables consistent comparisons and calculations.

---

8 Key & Identifier Creation

- Generated a **Location_Key** using:
- State + City + Zipcode
- Created a surrogate **Property_ID** using an index starting from 1001

Why: Supports clean relationships in the data model.

---

9 Enrichment via Lookup Merge

- Joined with a helper table on **City**
- Brought in **Median Price per Sq Ft**
- Recomputed MarketEstimate as:
- Median_Price_PSF × Area

Why: Produces a more stable market valuation baseline.

🔟 **Final Data Conditioning**

- Renamed fields for business readability
- Sorted records for consistency
- Final validation filters applied before load

DAX Measures — Organized & Explained

🔹 Core Pricing Metrics

Actual_ppsq

```
Actual_ppsq =
DIVIDE(
   SUM(Property_Fact[Price]),
   SUM(Property_Fact[Area])
)
```

Purpose:

Weighted average actual price per sq ft across properties.

✅ Correct aggregation logic (SUM/SUM, not AVERAGE).

Adjusted PPSq

```
Adjusted PPSq =
[Actual_ppsq] * (1 + 'Parameter'[Parameter Value])
```

Purpose:

Applies the What-If slider (–20% to +20%) to price per sq ft.

Used in Scenario Insights visuals.

MedianPricePerSqFt

MedianPricePerSqFt =

MEDIAN(Property_Fact[Price_per_Sqft])

Purpose:

Median pricing benchmark, less sensitive to outliers.

- Market Value Calculations

Adjusted Market Value

Adjusted Market Value =

SUM(Property_Fact[Area]) * [Adjusted PPSq]

Purpose:

Scenario-based market valuation driven by the What-If parameter.

Price_per_Sqft

Price_per_Sqft =

[Price] / [Area]

Purpose:

Row-level calculated metric used for variance and median analysis.

Price_Variance

Price_Variance =

([Price_per_Sqft] * [Area]) - [MarketEstimate]

Purpose:

Difference between actual pricing and estimated market value.

- Rental & Yield Metrics

Rental Yield

Rental Yield =

[RentEstimate] / [Price]

Purpose:

Property-level rental return ratio.

RentYield

```
RentYield =
DIVIDE(
    SUM(Property_Fact[RentEstimate]),
    SUM(Property_Fact[Price])
)
```

Purpose:

Portfolio-level rental yield (correctly weighted).

Adjusted Rental Yield

```
Adjusted Rental Yield =
DIVIDE(
    SUM(Property_Fact[RentEstimate]),
    [Adjusted Market Value]
)
```

Purpose:

Shows yield impact under pricing scenarios.

- Averages & Monitoring Metrics

Avg Price

```
Avg Price =
AVERAGE(Property_Fact[Price])
```

AvgPriceVariance

```
AvgPriceVariance =
AVERAGE(Property_Fact[Price_Variance])
```

AvgRentYield

```
AvgRentYield =
AVERAGE(Property_Fact[Rental Yield])
```

Purpose:

Used for trend cards, benchmarks, and executive summaries.

- Volume Metric

Total Properties

Total Properties =

DISTINCTCOUNT(Property_Fact[Property_ID])

Purpose:

Unique property count across filters.