

PERTH · DATA LIFECYCLE

Perth urban *sprawl*

Reading 13 years of traffic and population data.

What the numbers support, and what they do not.

Adina Lieblich

adinalieblich.com/wasprawl

Three *public* datasets



Main Roads WA

Loop detectors, 15-min volumes

2013 to 2026 · 106M rows

[Open source](#) ↗



ABS

Regional population, suburb and council

2013 to 2025 · SA2 and LGA

[Open source](#) ↗



Travel survey

Perth Area Travel & Household Survey (Main Roads WA)

Trips, distance and mode

[Open source](#) ↗

From raw files to *charts*



Downloaded, converted, loaded to Microsoft Fabric, queried, visualised.

In the *Fabric* lakehouse

The screenshot displays the Microsoft Fabric lakehouse explorer interface. The top navigation bar includes a search bar, a 'wa_traffic' dropdown, and a 'Power BI trial: 58 days left' notification. The left sidebar contains navigation icons for Home, Copilot, Create, OneLake catalog, Apps, Scorecards, Workspaces, WA Sprawl, and wa_traffic. The main content area shows the 'area_anim' table in 'Table view' mode, displaying 1000 rows. The table has 12 columns: ABC area, ABC corridor, year_month, 12L pop, 12 index_pct, 12 Lat, 12 Lon, 123 yr, and 123 mo. The data shows a sequence of records for the 'Alkimos - Eglinton' area, with population values ranging from 8171 to 8772 and dates from 2013-01-01 to 2014-01-01. A status bar at the bottom indicates 'Succeeded (1 sec 671 ms)' and 'Columns 12 Rows 1,000'.

wa_traffic

Search

Power BI trial: 58 days left

Home

Materialized lake views

Analyze data with

Share

Get data

New Spark SQL query

New semantic model

Add to data agent

Manage OneLake security

Update all variables

View, upload, and download lakehouse data directly from your desktop using the OneLake app.

Download OneLake app

Explorer

Add lakehouses

wa_traffic

Tables

dbo

area_anim

area_growth

control_traffic

corridor_proj...

corridor_traffic

detector_gro...

detector_trend

dim_segment

eastwest_dim

eastwest_traffic

events

mlinks_netwo...

area_anim

Table view

Showing 1000 rows

	ABC area	ABC corridor	year_month	12L pop	12 index_pct	12 Lat	12 Lon	123 yr	123 mo
1	Alkimos - Eglinton	mittchell_north_e...	2013-01-01T00:...	8171	100	-31.6	115.68	2013	1
2	Alkimos - Eglinton	mittchell_north_e...	2013-02-01T00:...	8221	100.6	-31.6	115.68	2013	2
3	Alkimos - Eglinton	mittchell_north_e...	2013-03-01T00:...	8271	101.2	-31.6	115.68	2013	3
4	Alkimos - Eglinton	mittchell_north_e...	2013-04-01T00:...	8321	101.8	-31.6	115.68	2013	4
5	Alkimos - Eglinton	mittchell_north_e...	2013-05-01T00:...	8371	102.5	-31.6	115.68	2013	5
6	Alkimos - Eglinton	mittchell_north_e...	2013-06-01T00:...	8421	103.1	-31.6	115.68	2013	6
7	Alkimos - Eglinton	mittchell_north_e...	2013-07-01T00:...	8472	103.7	-31.6	115.68	2013	7
8	Alkimos - Eglinton	mittchell_north_e...	2013-08-01T00:...	8522	104.3	-31.6	115.68	2013	8
9	Alkimos - Eglinton	mittchell_north_e...	2013-09-01T00:...	8572	104.9	-31.6	115.68	2013	9
10	Alkimos - Eglinton	mittchell_north_e...	2013-10-01T00:...	8622	105.5	-31.6	115.68	2013	10
11	Alkimos - Eglinton	mittchell_north_e...	2013-11-01T00:...	8672	106.1	-31.6	115.68	2013	11
12	Alkimos - Eglinton	mittchell_north_e...	2013-12-01T00:...	8722	106.7	-31.6	115.68	2013	12
13	Alkimos - Eglinton	mittchell_north_e...	2014-01-01T00:...	8772	107.4	-31.6	115.68	2014	1

Succeeded (1 sec 671 ms)

Columns 12 Rows 1,000

Every dataset loaded as a table in OneLake, browsable in the lakehouse explorer.

Loading the data with *PySpark*

The screenshot displays a Power BI notebook environment. The left sidebar shows the 'Explorer' pane with 'Data items', 'Resources', and 'Connections'. Under 'OneLake', a dataset named 'wa_traffic' is visible. The main editor area contains two code cells. The first cell contains a PySpark script that iterates through files in a directory, identifies parquet files, and loads them into a table. The second cell contains a PySpark script that reads a table named 'traffic_anim', transforms it, and saves it as 'area_anim'. The output of the second cell is visible, showing the columns of the 'traffic_anim' table.

```
1 # Welcome to your new notebook
2 # Type here in the cell editor to add code!
3 for fi in mssparkutils.fs.ls("Files/"):
4     if fi.name.endswith(".parquet"):
5         table = fi.name[:-8] # filename without ".parquet"
6         spark.read.parquet(f"Files/{fi.name}") \
7             .write.mode("overwrite").saveAsTable(table)
8         print("loaded:", table)
```

[1] ✓ - Session ready in 8 sec 561 ms. Command executed in 1 min 50 sec 641 ms by Adina LIEBLICH on 7/19/2026, 4:35:21 PM PySpark (Python) ✓

... Output is hidden

```
1 from pyspark.sql.functions import year, month, date_format
2 t = spark.read.table("traffic_anim").withColumn("yr", year("year_month")).withColumn("mo", month("year_mo
3 t.write.mode("overwrite").option("overwriteSchema", "true").saveAsTable("traffic_anim")
4 a = spark.read.table("area_anim").withColumn("yr", year("year_month")).withColumn("mo", month("year_mon
5 a.write.mode("overwrite").option("overwriteSchema", "true").saveAsTable("area_anim")
6 print("traffic cols:", spark.read.table("traffic_anim").columns)
7 print("done")
```

[1] ✓ PySpark (Python) ✓

traffic cols: ['MLinkID', 'grp', 'section', 'year_month', 'Lat', 'Lon', 'Road', 'vol_rolling12',
'vol_perlane_rolling12', 'index_raw', 'index_perlane', 'yr', 'mo', 'ym', 'ymlabel']
done

Not connected AutoSave: On Copilot completions: Off Selected Cell 1 of 16 cells

A notebook reads each parquet file and writes it as a lakehouse table.

STARTING POINT

The story I *expected*

ABS suburb figures single out one place, which frames the whole story as a single fringe-suburb boom.

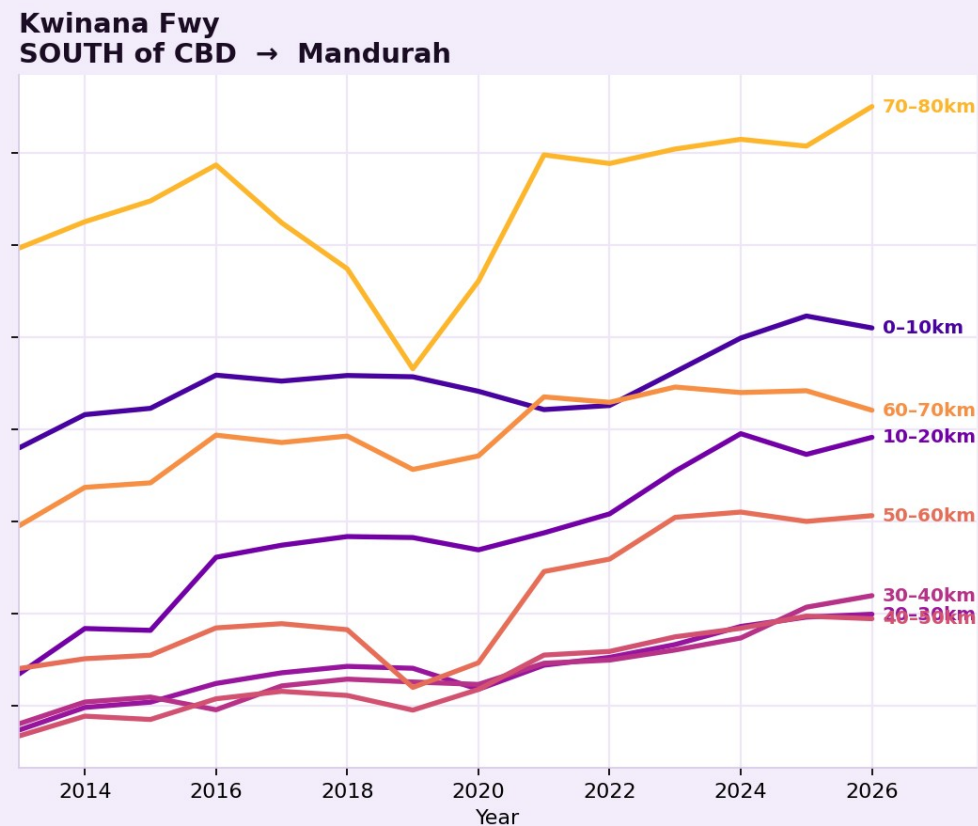
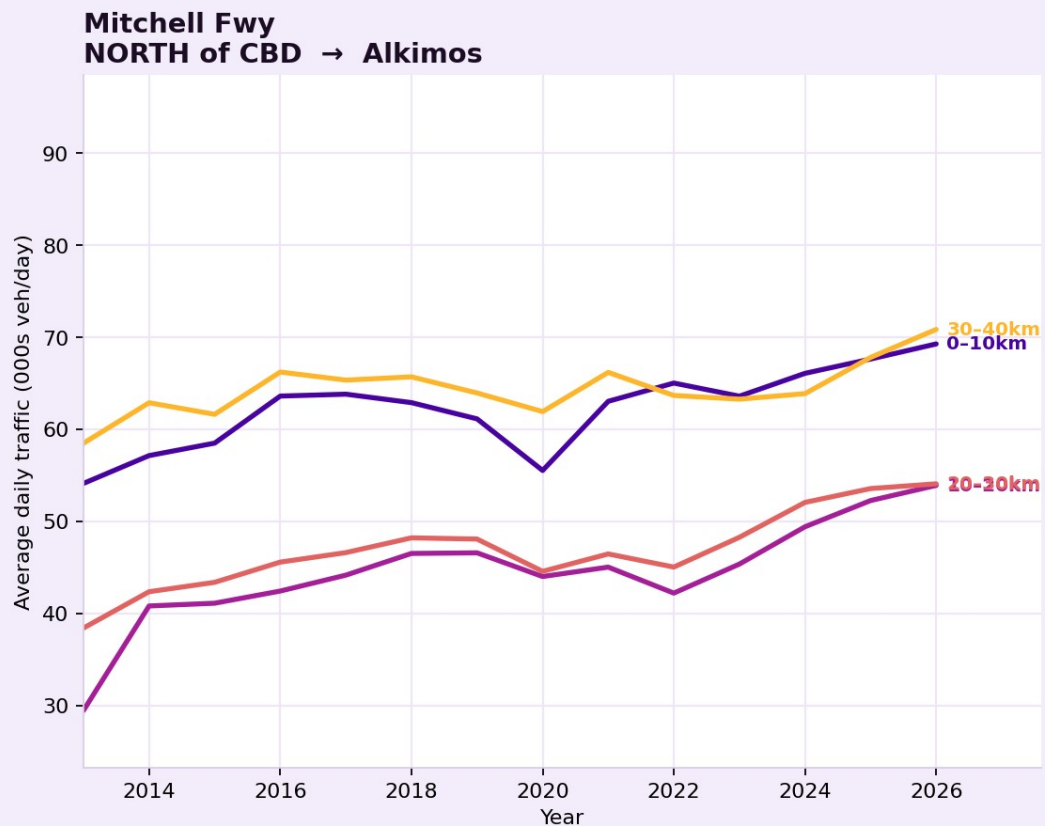
Alkimos-Eglinton grew 13% in one year, the fastest in Greater Perth (2.4%).

The test: did traffic concentrate there too, or spread across the corridor?

Traffic *grew* across the whole corridor

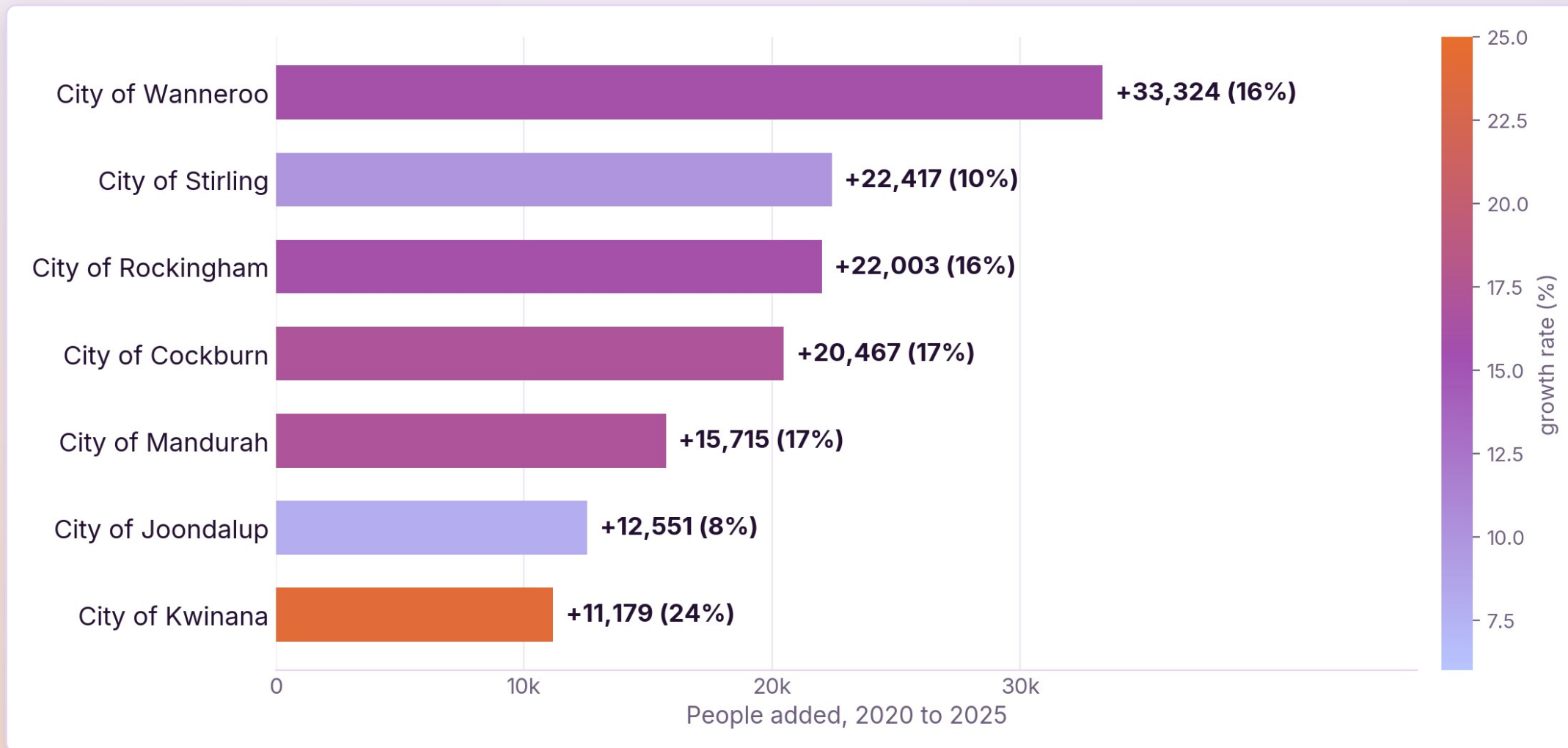
Raw traffic by 10 km band along the freeway — actual counts, no indexing, no curve-fitting

Average daily traffic per band, fixed set of established mainline detectors (same segments every year, so growth = real, not new road added). Each line = one 10 km stretch measured out from the CBD.



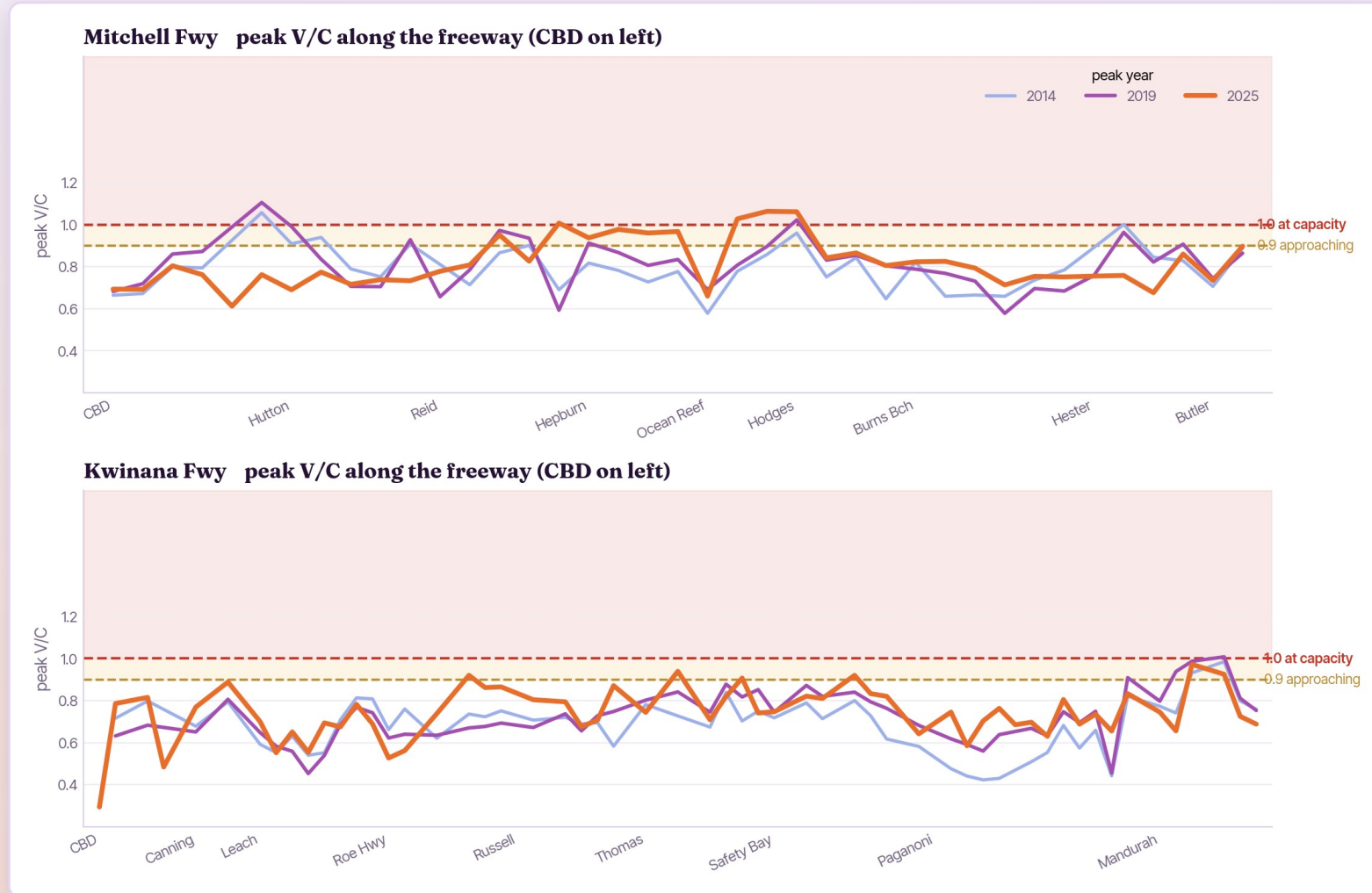
Raw average daily traffic, fixed detectors. No indexing, no curve fitting.

Growth by council: *people* vs *rate*



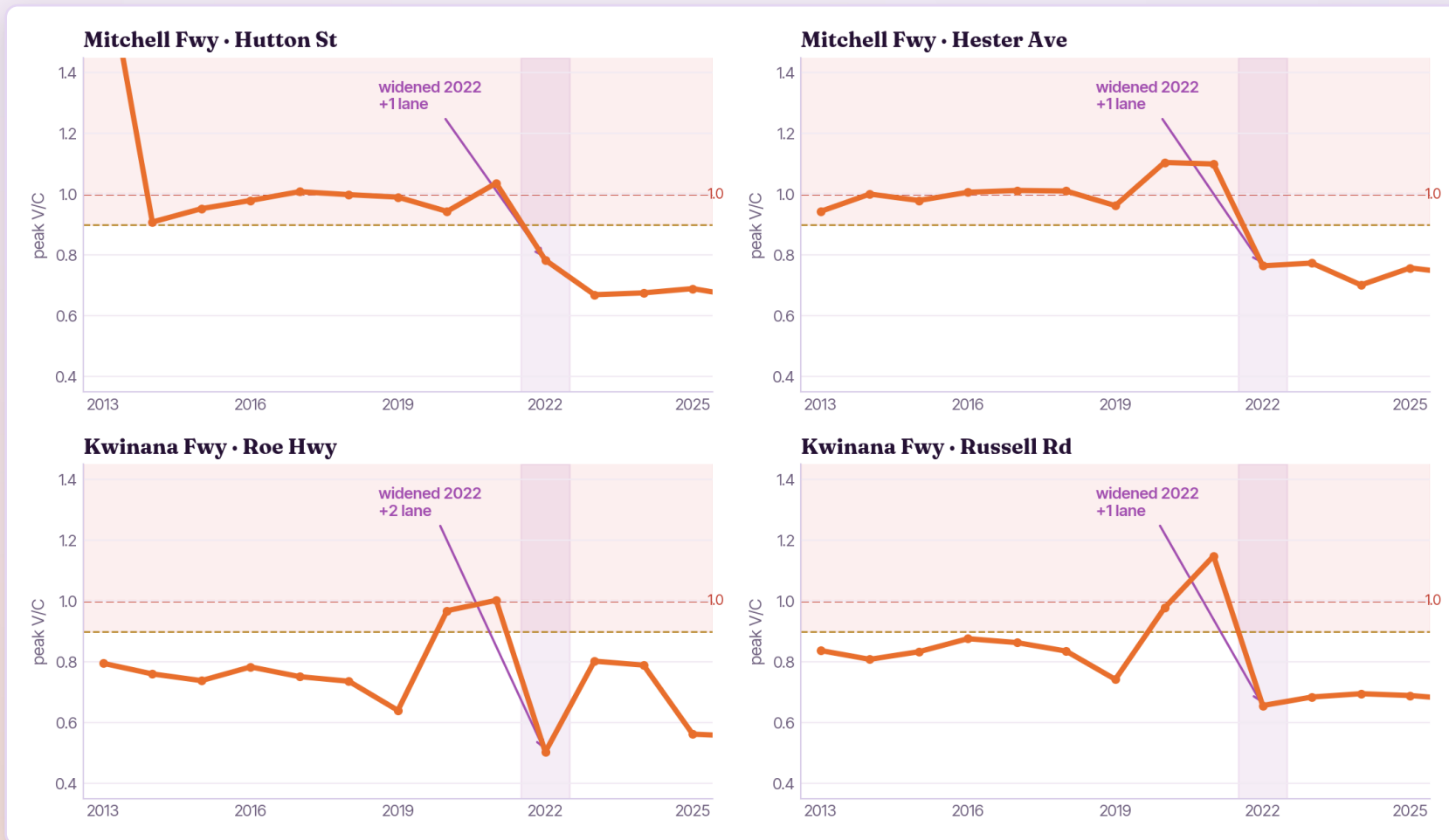
Bar length is people added. Colour is growth rate.

Where the freeway hits *capacity*



Peak-hour volume to capacity along each freeway. 0.9 approaching capacity, 1.0 at capacity.

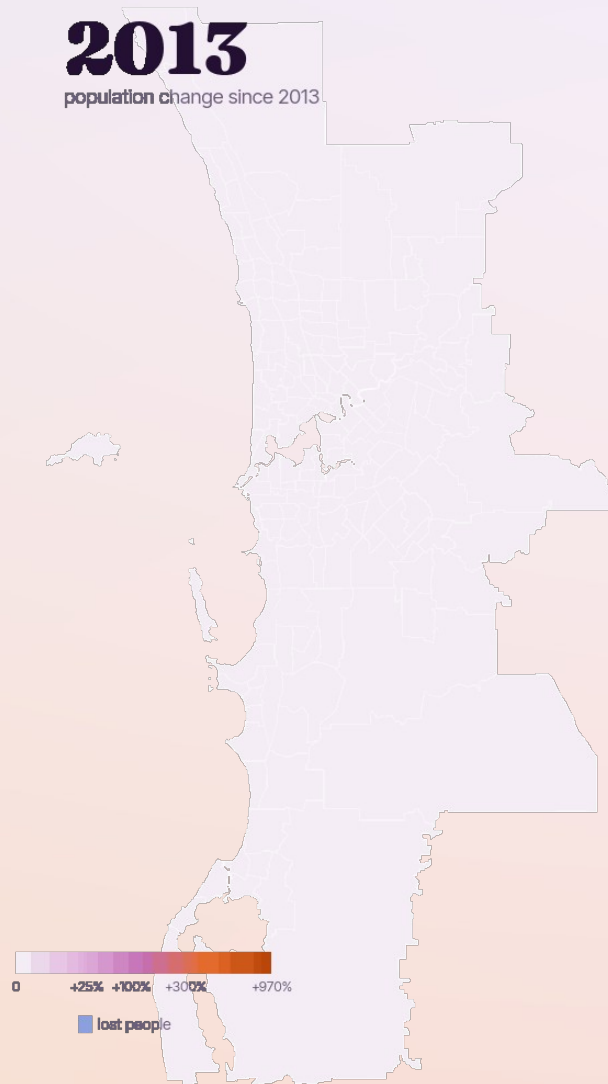
The dips are real *widenings*



Peak V/C climbs to capacity, then drops the exact year lanes were added (purple). The dips are real capacity, not noise.

Roe Hwy then refills within a year: new capacity filling back up, the induced-demand signature. Interchange labels approximate from chainage.

Population growth, *concentrated*



Selected suburbs, 2013 → 2023

Alkimos-Eglinton	1,655 → 17,706	+970%
Brabham-Henley Brook	4,201 → 14,337	+241%
Piara Waters	6,370 → 19,325	+203%
Baldivis North	5,993 → 14,158	+136%
Willetton	18,690 → 20,701	+11%
Ballajura	20,408 → 19,463	-5%

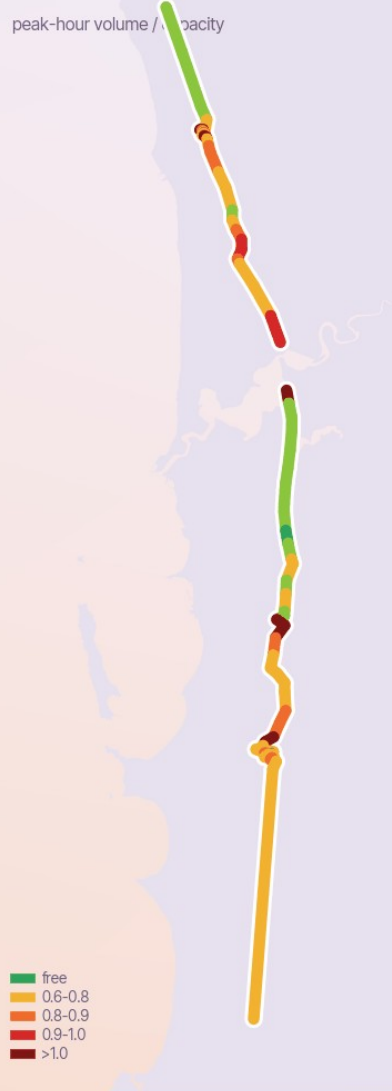
→ A ring of outer suburbs, not the whole city.

Map colour = growth since 2013; blue = lost people. Animates in slideshow (F5).

The freeway *fills up*

2013

peak-hour volume / capacity



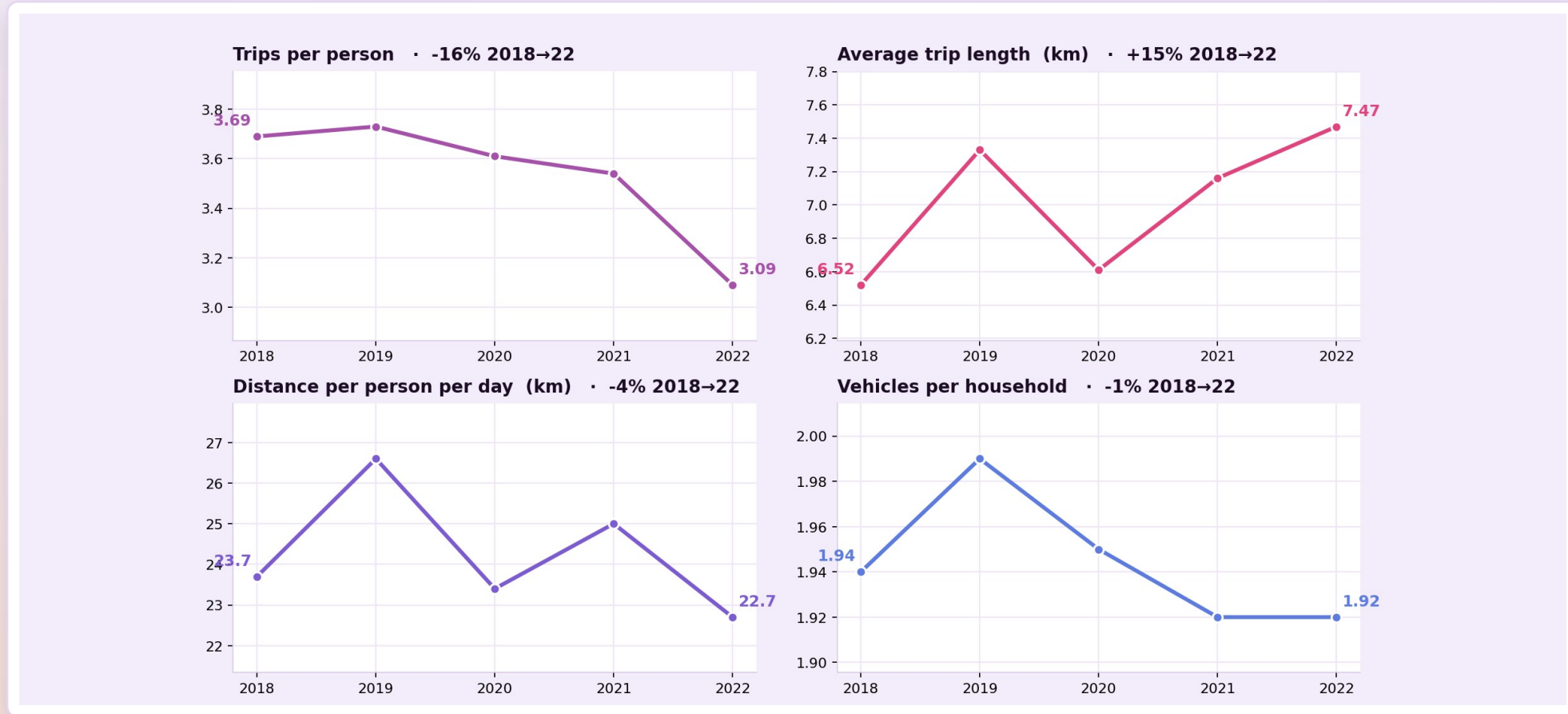
Each stretch is coloured by volume-to-capacity at peak hour, like a live traffic map: green is free-flowing, red is at or over capacity.

Year on year the whole corridor reddens, north and south, worst near the CBD and the Narrows.

→ **Congestion spreads along the whole corridor.**

Volume/capacity = peak-hour flow ÷ lane capacity. Animates in slideshow (F5).

How Perth *travels*, with caveats



Household travel survey, 2018 to 2022: trips per person fell 16% while the average trip got 15% longer.

Survey estimates, not a census: small samples (about 2,000 to 4,300 people a year), 2020 field work was disrupted by COVID, and the series ends before the 2023 freeway extension. Indicative, not proof.

THE POINT

What the data can, and *cannot*, say.

- Raw counts only. No smoothing, no indexing.
- Population growth is concentrated; traffic growth is corridor wide.
- Where the data ran out, I stopped short of the claim.