KPI Framework for Analysis

Summary

Key Performance Indicators (KPIs) to identify urban stress zones in Zurich by comparing traffic growth and population growth at the district level (2012–2025). The analysis supports the City of Zurich Urban Planning Office in prioritizing infrastructure investments.

Analysis Goals

Primary Goal

Identify **Stress Zones**: districts where population growth outpaces traffic capacity improvements, or vice versa.

Secondary Goal

Detect **Peak-Hour Bottlenecks**: specific locations experiencing sustained high traffic loads that warrant intervention.

Optional Goal

Directional Imbalance: MeasurementSite table contains a direction column (varchar) with values "inbound" and "outbound"; flow toward city center = "inbound", away = "outbound".

KPI 1: District-Level Stress Index

Definition

The **Stress Index** compares traffic volume growth rate vs. population growth rate for each district.

KPI 1.1: Aggregate Population by District & Year (important and yearly is simpler)

Source Tables: `Population`, `Quarter`
SQL Logic:

```sql
SELECT
    q.city_district,
    p.reference_date_year,
    SUM(p.population_count) AS total_population
FROM Population p
JOIN Quarter q ON p.statistical_quarter = q.statistical_quarter
WHERE p.reference_date_year IN (2012, 2025)
GROUP BY q.city_district, p.reference_date_year;
```

Output: Population per district for start (2012) and endpoint (2025).

KPI 1.2: Aggregate Traffic Volume by District & Year

Source Tables: `TrafficMeasurement, MeasurementSite, CountingSite, Quarter`
SQL Logic:

```sql
SELECT
    q.city_district,
    YEAR(tm.timestamp) AS year,
    SUM(tm.vehicle_count) AS total_vehicle_count
FROM TrafficMeasurement tm
JOIN MeasurementSite ms ON tm.measurement_site_id = ms.measurement_site_id
JOIN CountingSite cs ON ms.counting_site_id = cs.counting_site_id
JOIN Quarter q ON cs.axis = q.street_name  -- assumes axis maps to street_name
WHERE YEAR(tm.timestamp) IN (2012, 2025)
  AND tm.vehicle_count_status = 'Measured'
GROUP BY q.city_district, YEAR(tm.timestamp);
```

Output: Total vehicle counts per district for 2012 and 2025.

KPI 1.3: Calculate Growth Rates

Calculation: - **Population Growth (%)** = `((Pop_2025 - Pop_2012) / Pop_2012) × 100` - **Traffic Growth (%)** = `((Traffic_2025 - Traffic_2012) / Traffic_2012) × 100`

KPI 1.4: Compute Stress Index

Formula:

`Stress Index = Traffic Growth % - Population Growth %`

Interpretation: - **Positive Stress Index**: Traffic growth > Population growth → **Commuter Hub** (inbound pressure) - **Negative Stress Index**: Population growth > Traffic growth → **Residential Zone** (capacity lag) - **Near Zero**: Balanced growth

KPI 1.5: Classify Districts

Thresholds: - `Stress Index > +10%` → **High Commuter Pressure** - `Stress Index < -10%` → **High Residential Pressure** - `-10% <= Stress Index <= +10%` → **Balanced**

Output Table:

District	Pop Growth %	Traffic Growth %	Stress Index	Classification
1	15.2%	8.3%	-6.9%	Balanced
2	12.5%	22.1%	+9.6%	Commuter Hub
...	...	...	...	...

KPI 2: Peak-Hour Bottleneck Identification *(only if we need more analysis)*

Definition

Identify **specific counting sites** (not aggregated by district) where peak-hour traffic consistently exceeds capacity thresholds.

Calculation Steps

KPI 2.1: Calculate Average Hourly Traffic per Counting Site

Source Tables: `TrafficMeasurement`, `MeasurementSite`, `CountingSite`
SQL Logic:

```sql
SELECT
    cs.counting_site_id,
    cs.counting_site_name,
    HOUR(tm.timestamp) AS hour_of_day,
    AVG(tm.vehicle_count) AS avg_hourly_volume
FROM TrafficMeasurement tm
JOIN MeasurementSite ms ON tm.measurement_site_id = ms.measurement_site_id
JOIN CountingSite cs ON ms.counting_site_id = cs.counting_site_id
WHERE tm.vehicle_count_status = 'Measured'
  AND YEAR(tm.timestamp) BETWEEN 2023 AND 2025  -- recent 3 years
GROUP BY cs.counting_site_id, cs.counting_site_name, HOUR(tm.timestamp);
```

KPI 2.2: Identify Peak Hour per Site *(based on counting_site, e.g., Seestrasse (Wollishofen))*

Logic: For each counting site, select the hour with maximum `avg_hourly_volume`.

KPI 2.3: Apply Bottleneck Threshold

Threshold: `avg_hourly_volume > 800 vehicles/hour` (sustained pressure)

Output Table:

Counting Site Name	Peak Hour	Avg Volume	Status
Hardbrücke	08:00	1,245	Bottleneck
Bellevue	17:00	1,102	Bottleneck
Seestrasse (Wollishofen)	07:00	650	Normal

KPI 3: Directional Imbalance (Optional, based on counting_site)

Definition

Ratio of inbound vs. outbound traffic at key counting sites, indicating unbalanced flow.

Calculation

```sql
SELECT
    cs.counting_site_name,
    SUM(CASE WHEN ms.direction = 'inbound' THEN tm.vehicle_count ELSE 0 END) AS inbound_total,
    SUM(CASE WHEN ms.direction = 'outbound' THEN tm.vehicle_count ELSE 0 END) AS outbound_total,
    ROUND(
        SUM(CASE WHEN ms.direction = 'inbound' THEN tm.vehicle_count ELSE 0 END) /
        NULLIF(SUM(CASE WHEN ms.direction = 'outbound' THEN tm.vehicle_count ELSE 0 END), 0),
    2) AS directional_ratio
FROM TrafficMeasurement tm
JOIN MeasurementSite ms ON tm.measurement_site_id = ms.measurement_site_id
JOIN CountingSite cs ON ms.counting_site_id = cs.counting_site_id
```

```
WHERE YEAR(tm.timestamp) = 2025
GROUP BY cs.counting_site_name
HAVING directional_ratio > 1.5 OR directional_ratio < 0.67;
```

Threshold: Ratio > 1.5 or < 0.67 indicates **strong imbalance**.

KPI 4: Dashboard & Visualization Plan

KPI 4.1: Visual 1: District Stress Index Map

- **Type**: Choropleth map of Zurich districts
- **Color Scale**: Red (high residential pressure) → Yellow (balanced) → Blue (high commuter pressure)
- **Data**: KPI 1 output

KPI 4.2: Visual 2: Peak-Hour Bottleneck Heatmap

- **Type**: Time-of-day heatmap (rows = counting sites, columns = hours)
- **Color Scale**: Green (low traffic) → Red (bottleneck threshold exceeded)
- **Data**: KPI 2 output

KPI 4.3: Visual 3: Growth Trend Comparison

- **Type**: Grouped bar chart (12 districts × 2 bars: population growth vs. traffic growth)
- **Data**: KPI 1 intermediate results (Step 1.3)

Any Other Visualization (to be selected during analysis)

KPI 5: Actionable Insights for Planning Decisions

Definition

Translate KPI 1–4 outputs into prioritized recommendations for urban planning strategies.

KPI 5.1: District Priorities

Input: Stress Index classification (from KPI 1)

Output: Ranked list of **districts** by urgency

Priority	Stress Index Range	Recommended Action
High	< -15% (residential pressure)	Expand public transit, add lanes, optimize signals
High	> +15% (commuter pressure)	Improve inbound capacity, park-and-ride facilities
Medium	-15% to -10% or +10% to +15%	Monitor trends, schedule reassessment in 2 years
Low	-10% to +10% (balanced)	Maintain current infrastructure

KPI 5.2: Site-Level Interventions

Input: Bottleneck sites (from KPI 2)

Actions: - **Top 5 bottleneck sites (streets)**: immediate signal timing optimization or capacity studies - **Sites (streets) with directional imbalance > 1.5** (from KPI 3): directional lane adjustments, reversible lanes

KPI 5.3: Budget Allocation Guide

Combine district-level (KPI 1) and site-level (KPI 2) findings:

1. **Short-term budget (1–2 years)**: Fix top 10 bottleneck intersections
2. **Long-term budget (3–5 years)**: Address high-stress districts with infrastructure projects

---