SkyAware TEMPO API - Data Coverage & Usage Guide

Current Data Coverage

Geographic Extent (Current Snapshot)

Based on latest TEMPO data ingestion:

```
Latitude Range:  24.01°N to 24.61°N
Longitude Range: -122.05°W to -65.01°W
Total Points:    85,000
Last Updated:    2025-10-05 14:53:56 UTC
```

Coverage Area: Southern California/Mexico border region to East Coast (but limited latitude range)

Why Limited Coverage?

TEMPO satellite data coverage varies by: 1. **Time of Day** - TEMPO only observes during daylight (hourly from ~8am-6pm local time) 2. **Cloud Cover** - Areas with clouds are filtered out for data quality 3. **Satellite Position** - Geostationary orbit provides full longitude coverage but scan angle affects latitude 4. **Quality Flags** - Only high-quality data (quality_flag=0) is processed

API Usage Examples

✅ Working Example (Data Available)

```
# Southern California/Mexico border area (24.3°N, 118°W)
curl "https://tempo-api-336045066613.us-central1.run.app/latest-aqi?lat=24.3&lon=-118.0&radius=100&limit=10"
```

Response (3-5 seconds):


```
{
  "source": "database",
  "sampled": true,
  "processed_points": 5000,
  "matches": 174,
  "data": [
    {
      "latitude": 24.29,
      "longitude": -118.01,
```

```
      "aqi": 146.0,
      "category": ["Unhealthy for Sensitive Groups", "#FF7E00"],
      "distance_km": 1.5,
      "timestamp": "2025-10-05T14:53:56Z"
    }
  ]
}
```


✗ Out of Range Example

```
# Los Angeles (34.05°N, 118.25°W) - Outside current data range
curl "https://tempo-api-336045066613.us-central1.run.app/latest-aqi?
lat=34.05&lon=-118.25&radius=50"
```

Response:


```
{
  "error": "No data found within 50km of specified location (sampled 5000 points)"
}
```


Finding Available Data

Method 1: Query Without Location

Get general data and see what's available:

```
curl "https://tempo-api-336045066613.us-central1.run.app/latest-aqi?limit=10"
```

This returns the first 10 points from the dataset, showing you actual coordinates.

Method 2: Use Wide Radius

Start with a large radius to find nearby data:

```
curl "https://tempo-api-336045066613.us-central1.run.app/latest-aqi?
lat=30.0&lon=-100.0&radius=1000&limit=20"
```

Method 3: Check Database Directly

```
import psycopg2
conn = psycopg2.connect(
    host='34.134.159.215',
    port=5432,
    user='tempo_user',
    password='Tempo_P@ss2443',
    database='tempo_aqi_db'
)
cursor = conn.cursor()
cursor.execute("""
    SELECT
        MIN((data->0->>'latitude')::float) as min_lat,
        MAX((data->0->>'latitude')::float) as max_lat,
        MIN((data->0->>'longitude')::float) as min_lon,
        MAX((data->0->>'longitude')::float) as max_lon
    FROM tempo_aqi
    WHERE timestamp = (SELECT MAX(timestamp) FROM tempo_aqi)
""")
print(cursor.fetchone())
```

API Parameters

GET /latest-aqi

Parameter	Type	Required	Default	Description

`lat`	float	No	-	Latitude for location search
`lon`	float	No	-	Longitude for location search
`radius`	float	No	50	Search radius in kilometers
`limit`	int	No	100	Maximum results to return

Notes: - If `lat`/`lon` provided: Returns points within radius, sorted by distance - If `lat`/`lon` omitted: Returns general data sample (up to limit) - Coordinates use WGS84 (standard GPS coordinates) - Negative longitude = West, Positive = East - Positive latitude = North, Negative = South

Response Fields

```
{
  "source": "database" | "redis_cache",
  "sampled": true,
  "processed_points": 5000,
  "matches": 174,
  "data": [
    {
      "latitude": 24.29,           // Degrees North
      "longitude": -118.01,         // Degrees West
      "aqi": 146.0,                 // EPA AQI value (0-500)
      "category": [                 // AQI category and color
        "Unhealthy for Sensitive Groups",
        "#FF7E00"
      ],
      "distance_km": 1.5,           // Distance from query point (km)
      "location": "TEMPO_24.29_-118.01",  // Location identifier
      "no2_concentration": 205024487753413.88,  // NO2 concentration (molecules/cm²)
      "timestamp": "2025-10-05T14:53:56Z"
    }
  ]
}
```

AQI Categories

AQI Range	Category	Color	Health Implications
0-50	Good	Green (#00E400)	Air quality is satisfactory
51-100	Moderate	Yellow (#FFFF00)	Acceptable for most people
101-150	Unhealthy for Sensitive Groups	Orange (#FF7E00)	Sensitive groups may experience health effects
151-200	Unhealthy	Red (#FF0000)	Everyone may begin to experience health effects
201-300	Very Unhealthy	Purple (#8F3F97)	Health alert: everyone may experience more serious health effects
301-500	Hazardous	Maroon (#7E0023)	Health warning of emergency conditions

Performance

Current (Database Mode)

- **Response Time**: 3-5 seconds
- **Processing**: Samples 5,000 points intelligently
- **Accuracy**: High (finds nearest points reliably)

Future (With Redis Cache)

- **Response Time**: <50 milliseconds
- **Processing**: Pre-computed results in memory
- **Accuracy**: Exact (no sampling needed)

Troubleshooting

“No data found within Xkm”

Cause: Query location outside current data coverage
Solution: 1. Check current coverage area (latitude 24-25°N currently) 2. Increase radius parameter 3. Try coordinates within known coverage area 4. Query without lat/lon to see available data

“Response timeout” or very slow

Cause: Redis cache not connected, falling back to database
Solution: 1. Current performance (3-5s) is acceptable for database mode 2. For <50ms: Configure VPC Connector for Redis access 3. Increase timeout in client if needed

“Database connection failed”

Cause: Database temporarily unavailable
Solution: Retry after a few seconds

Example Integration (JavaScript)

```
        async function getNearbyAQI(latitude, longitude, radiusKm = 50) {
          const url = `https://tempo-api-336045066613.us-central1.run.app/latest-aqi?
lat=${latitude}&lon=${longitude}&radius=${radiusKm}&limit=10`;

          try {
            const response = await fetch(url);
            const data = await response.json();

            if (data.error) {
              console.log('No data in this area:', data.error);
              return null;
            }

            console.log(`Found ${data.matches} points within ${radiusKm}km`);
            console.log(`Nearest: ${data.data[0].distance_km}km away, AQI: ${data.data[0].aqi}`);

            return data;
          } catch (error) {
            console.error('API error:', error);
            return null;
          }
        }

        // Example: Query southern California
        getNearbyAQI(24.3, -118.0, 100);
```

Data Updates

- **Frequency**: Hourly (TEMPO satellite measurement frequency)
- **Latency**: ~30 minutes from satellite observation to API availability
- **Retention**: Latest snapshot only (not historical data)
- **Schedule**: Automated Cloud Scheduler runs pipeline hourly

Support

For questions about data coverage or API usage: 1. Check `PERFORMANCE_STATUS.md` for system status 2. Review `REDIS_CACHING_OPTIMIZATION.md` for architecture 3. Run `python3 test_redis_cache.py` to test connectivity

---

Last Updated: October 5, 2025
API Version: v1
Status: ✅ Operational (Database mode, 3-5s responses)