

# Time Series Databases

Lorna Mitchell, Aiven



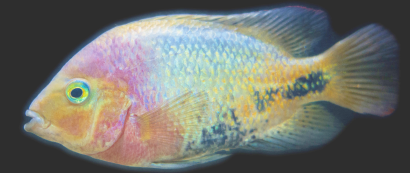
# Time Series Data

- Timestamp information
- Point-in-time value
- Metadata (such as location, measurement type)
- Immutable



# Time Series Data

```
=====
Time      Temperature      Metadata
=====
11:17    24.8                 Fish tank
11:22    24.8                 Fish tank
11:27    25.0                 Fish tank
11:32    24.9                 Fish tank
=====
```



# How to work with Time Series Data in PHP

# How to work with Time Series Data in PHP

Use PostgreSQL



# Dataset

Weather in Manchester  
1980-2020



@lornajane

# Time Series Queries

With this much data, aggregation is very useful

Select the average temperature for each week with PostgreSQL:

```
select date_trunc('week', time) as week, avg(temp)
from weather group by week;
```

# Next-Level: TimescaleDB

"TimescaleDB is a category-defining relational database for time-series data."

- <https://www.timescale.com/>
- A PostgreSQL extension
- Hypertable support
- Walks and talks like PostgreSQL



# TimescaleDB Time\_Bucket

The `time_bucket()` function allows more flexible queries

- Query by second/minute/day/week as before
- Also by multiples of those, e.g. 5 minutes or 8 hours

```
select time_bucket('4 weeks', time) as month, avg(temp)
from weather group by month;
```

# TimescaleDB Aggregation

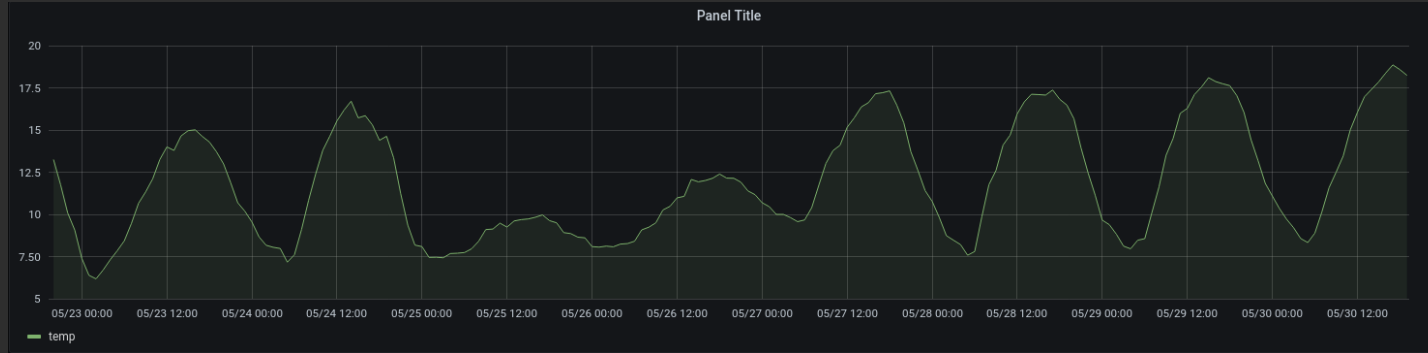
- Use downsampling to store reduced resolution data for longer periods
- TimescaleDB has continuous aggregates that define:
  - the time bucket size
  - which aggregates (average, maximum) to store
- Comparable to a materialised view

# Downsampling Data

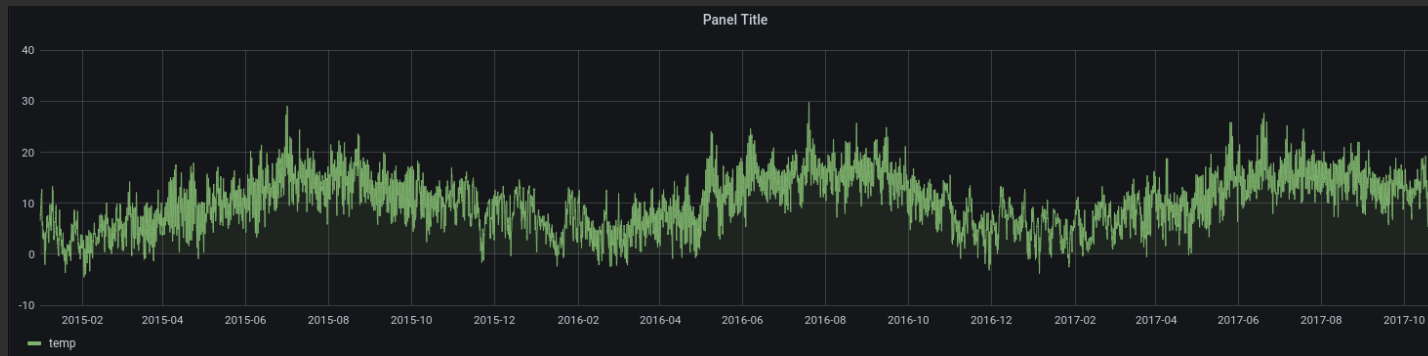


# Downsampling Data

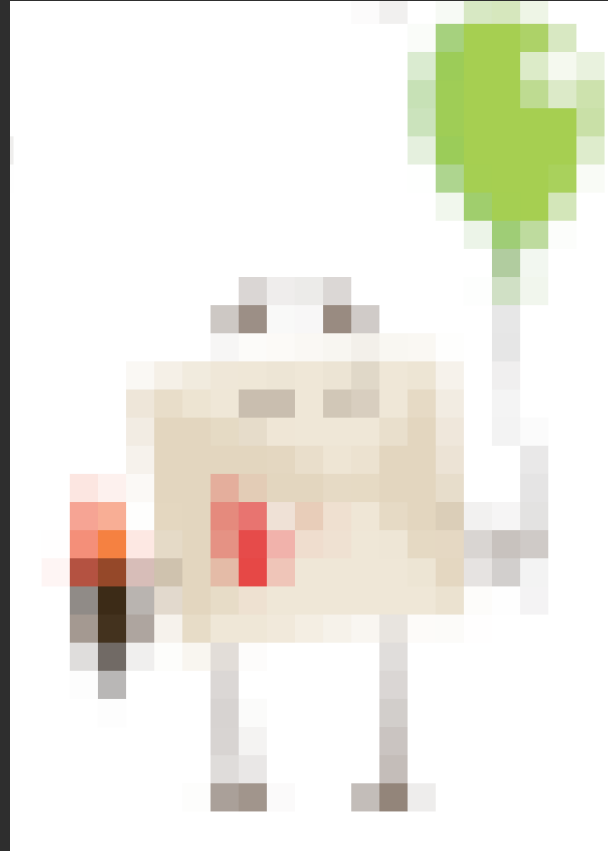
We don't need every datapoint, forever.



Above: 9 days. Below: 3 years.



# Reducing Detail Over Time

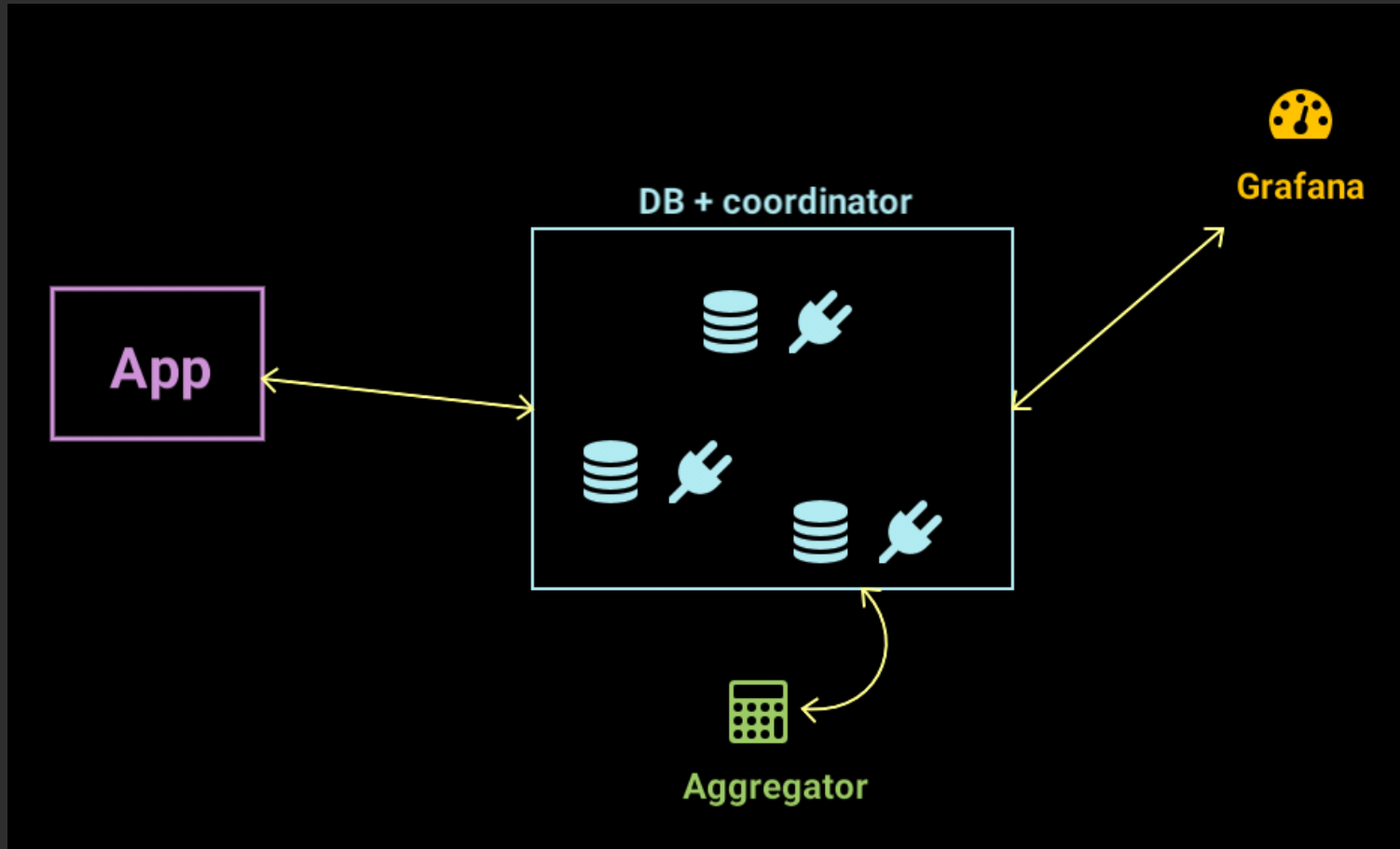


# **Specialist Time Series Databases**

# Specialist Databases

- Prometheus, open source metrics collector
- InfluxDB, open source for single node
- M3DB, open source, clustered
- VictoriaMetrics, open source, clustered

# M3DB Architecture





# PHP and M3DB

Influx DB wire protocol is supported:

```
composer require influxdb/influxdb-php
```

Connect to the database:

```
$client = new \InfluxDB\Client(  
    "https://example-service-name.aivencloud.com",  
    "13041",  
    "avnadmin",  
    "supersecretandsecure",  
    true,  
    true,  
);
```

# PHP and M3DB

## Sending data:




```
// construct a data point
$point = new \InfluxDB\Point(
    'php_example_metric', // name of the measurement
    0.64, // the measurement value
    ['host' => 'server1', 'location' => 'EU-DE-22'], // optional tags
    ['cpucount' => 8], // optional additional fields
);

// write to the database using the path
$result = $client->write(["url" => "api/v1/influxdb/write?db=default"], $point)
```

Pass as many points as you like!

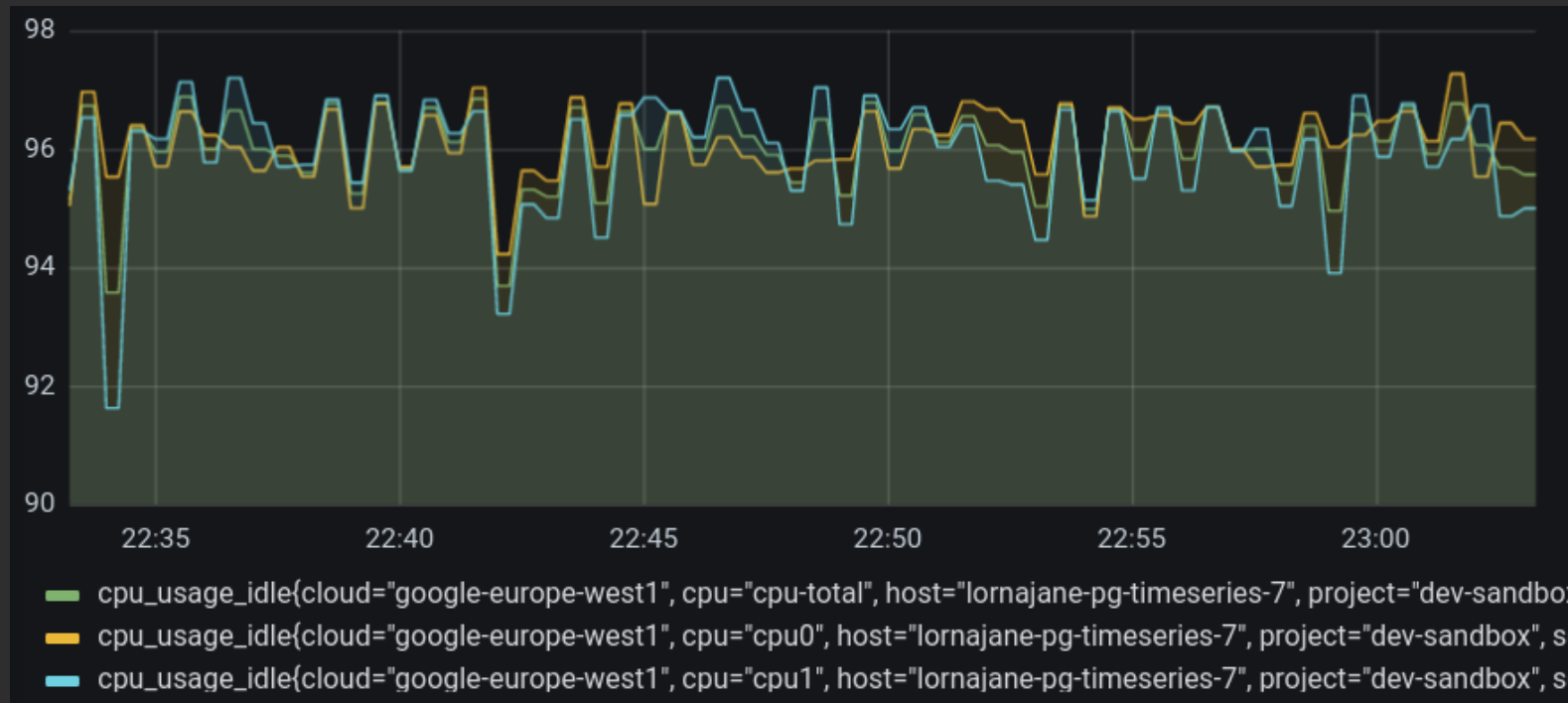
# Ready-Made Integrations

Try M3 by monitoring a services, view with Grafana

Service	Nodes	Plan	Cloud	Created
 <b>grafana-demo</b> Grafana • Running	●	<b>Startup-1</b> 1 CPU / 2 GB RAM	Google Cloud: europe-west3 Europe, Germany	4 minutes
 <b>m3db-demo</b> M3DB • Running	● ● ●	<b>Business-8</b> 2 CPU / 8 GB RAM / 450 GB storage - 3-node high availability set	Google Cloud: europe-west3 Europe, Germany	4 minutes
 <b>postgres-demo</b> PostgreSQL • Running	● ●	<b>Business-4</b> 1 CPU / 4 GB RAM / 80 GB storage - high availability pair	Google Cloud: europe-west3 Europe, Germany	4 minutes

# Ready-Made Integrations

Visualise server metrics with Grafana



# M3DB Namespaces

M3DB namespaces can have different retention and resolution of the data stored

- One unaggregated namespace, required
- Optional additional aggregated namespaces
  - resolution (e.g. 4h)
  - retention period (e.g. 30 days)

# M3DB Aggregator

Makes the aggregation magic work in the distributed system.

An optional (but recommended) extra component

- samples the incoming unaggregated data
- writes the aggregated results to M3DB

# **Time Series Databases**

**Specialist tools for specialist problems**

# Resources

- <https://aiven.io> (free trial!)
- <https://www.postgresql.org>
- <https://www.timescale.com>
- <https://www.influxdata.com>
- <https://m3db.io>
- <https://lornajane.net>

