TimescaleDB 2.0

Time-series данные
в распределенном кластере TimescaleDB
поверх ОРСУБД PostgreSQL

PGConf.Russia 2021
Иван Муратов @ binakot
25 – 26 октября | Москва

Спикер

Иван Муратов
CTO @ Waliot
https://github.com/binakot

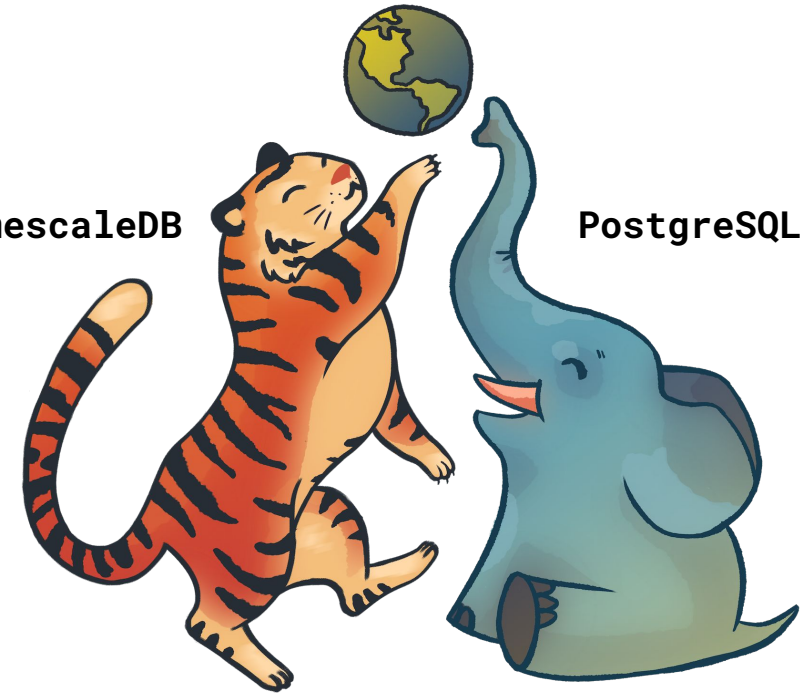
KRASNODAR DEV DAYS

Первая Мониторинговая Компания

KRASNODAR BACKEND

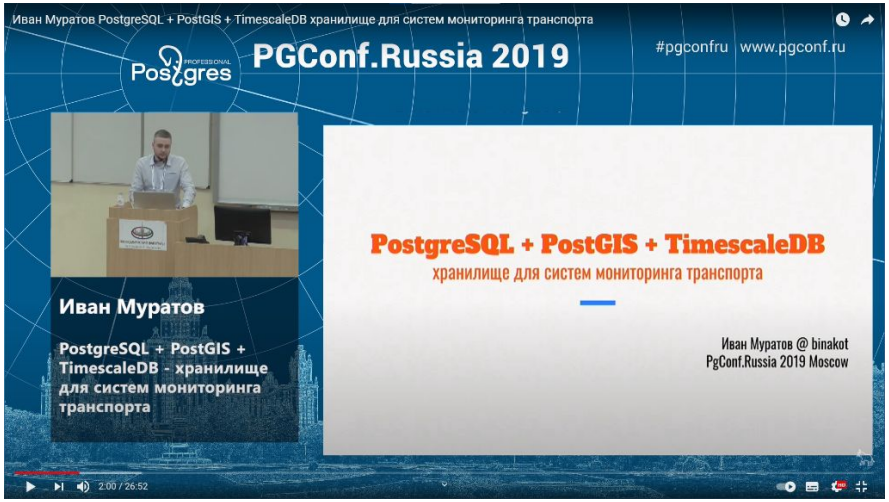
WALIOT

PostGIS

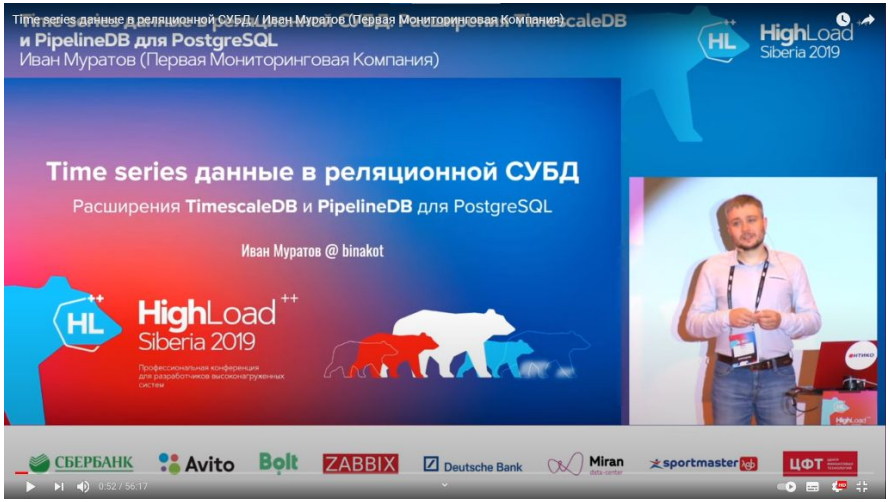


WALIOT

В предыдущих сериях...

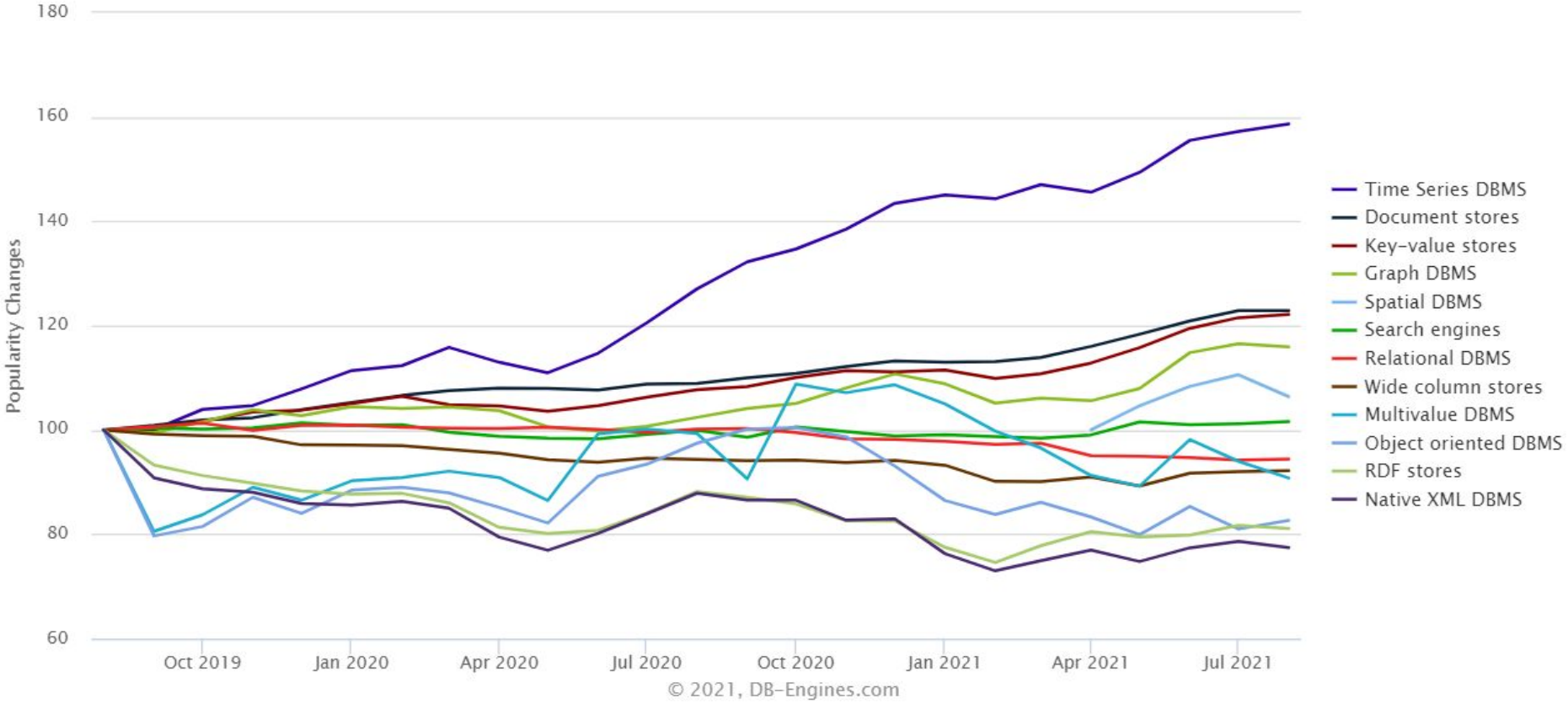


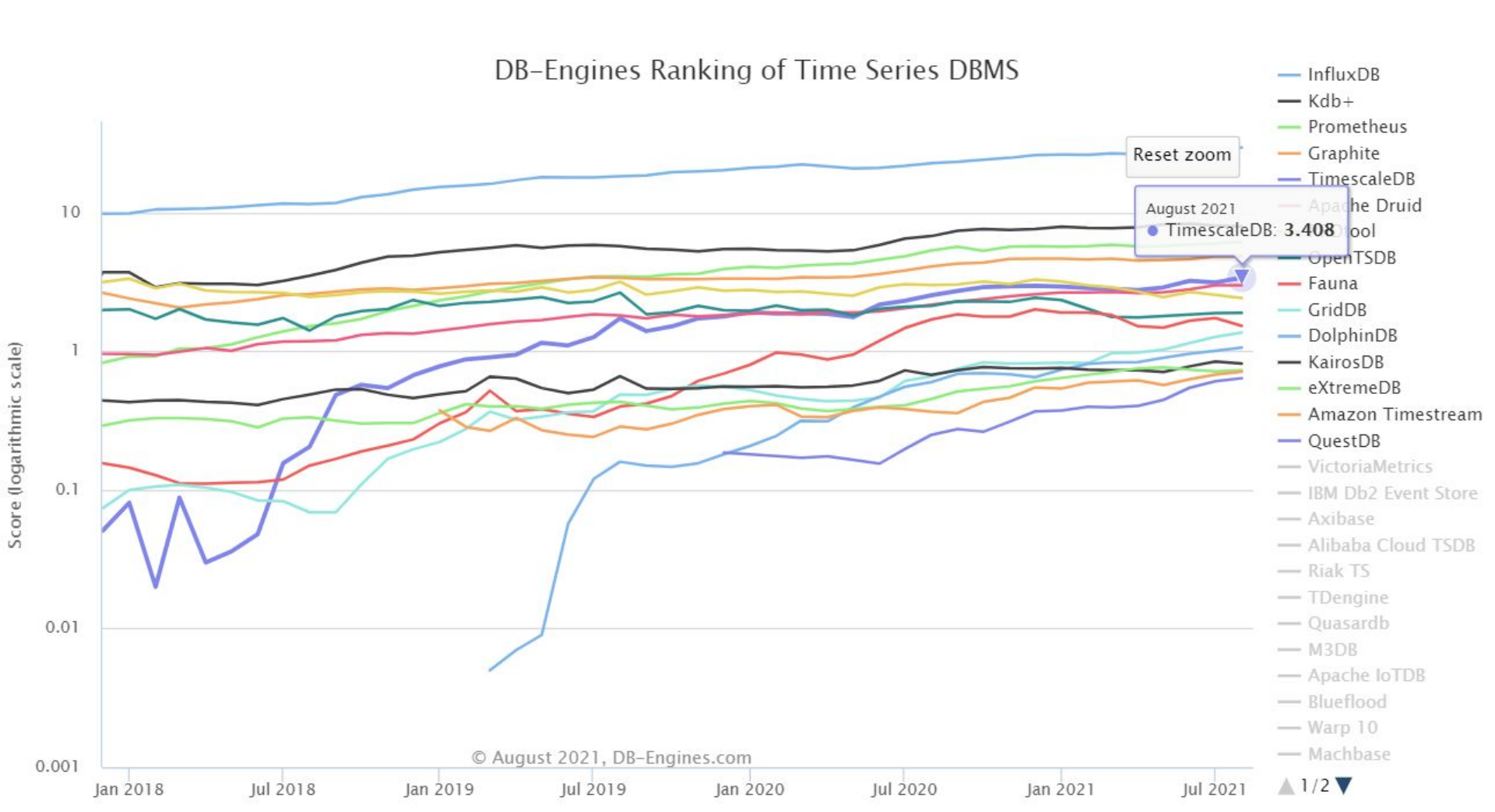
https://youtu.be/SEncf-h4Npw



https://youtu.be/3WkNp7mllv0

Trend of the last 24 months




DB-Engines Ranking of Time Series DBMS
Reset zoom
August 2021
TimescaleDB: 3.408
Score (logarithmic scale)
10
1
0.1
0.01
0.001
Jan 2018
Jul 2018
Jan 2019
Jul 2019
Jan 2020
Jul 2020
Jan 2021
Jul 2021
© August 2021, DB-Engines.com
InfluxDB
Kdb+
Prometheus
Graphite
TimescaleDB
Apache Druid
OpenTSDB
Fauna
GridDB
DolphinDB
KairosDB
eXtremeDB
Amazon Timestream
QuestDB
VictoriaMetrics
IBM Db2 Event Store
Axibase
Alibaba Cloud TSDB
Riak TS
TDengine
Quasardb
M3DB
Apache IoTDB
Blueflood
Warp 10
Machbase
1/2

Time series

Временные ряды — собранные в разные моменты времени данные о характеристиках исследуемого процесса.

Временные ряды состоят из двух элементов:

1. временная метка;
2. характеристики, называемые уровнями ряда.

Особенности time-series данных:

- Time-centric
- Append-only
- Recent

postgresmen
@postgresmen

Following

One SQL to Rule Them All: Management of Streams and Tables
arxiv.org/abs/1905.12133

▲ truth_seeker 10 hours ago [-]

Great write-up. SQL is the most underrated language.

I am doing most of the stuff mentioned in that paper right now in my project with PipelineDB extension.

PostgreSQL 11 +

CitusDB extension (horizontal scaling and sharding) +

cstore_fdw extnesion (columnar data storage) +

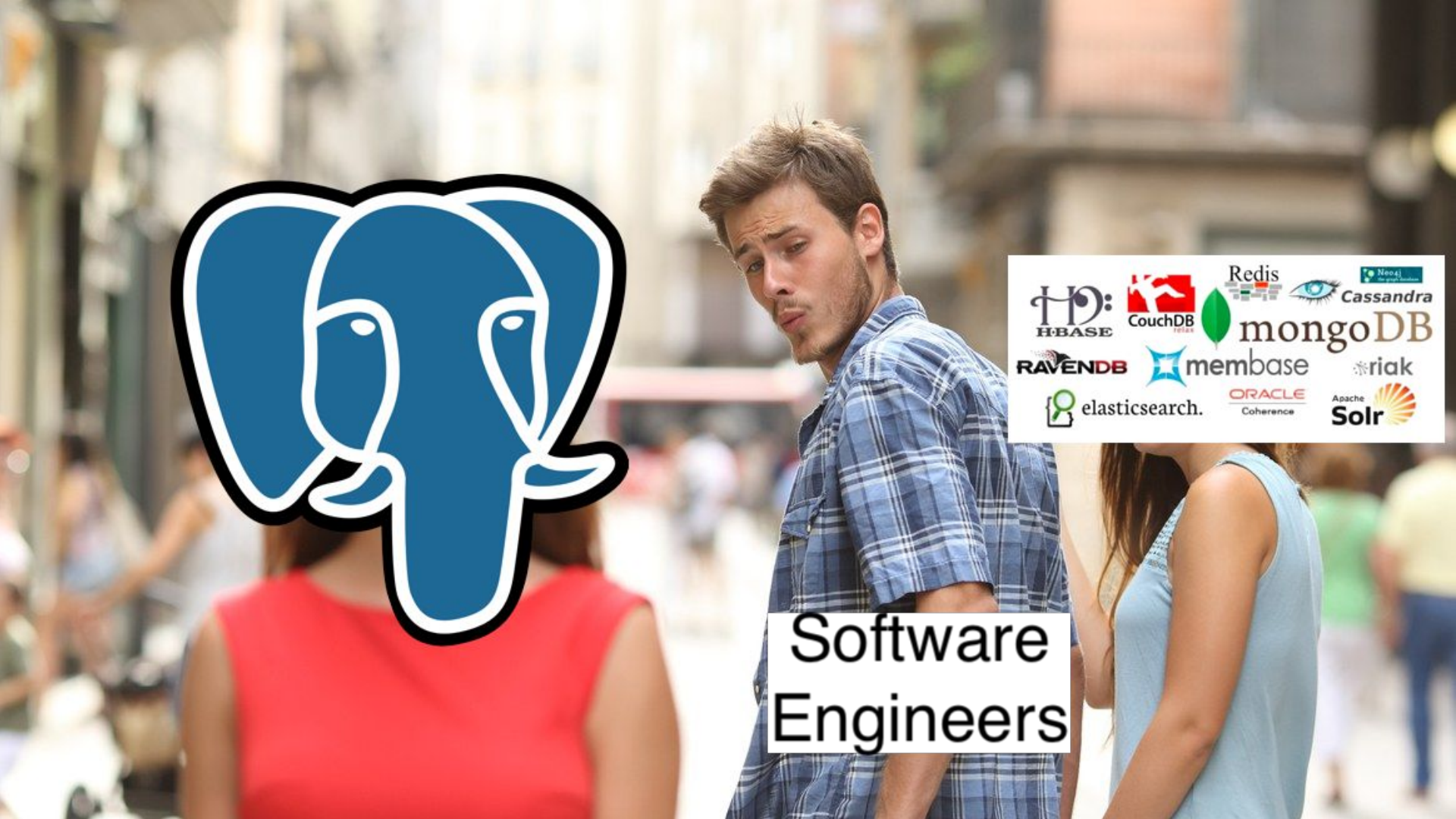
file_fdw (to treat file as table) +

PipelineDB extension (Streaming computations and MV) +

TimescaleDB extension (timeseries data storage)

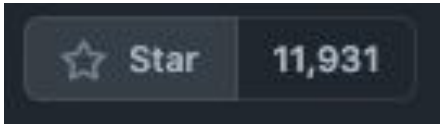
Also, PG 12 release will have many optimizations for table partitioning, query planning and its parallel execution

https://news.ycombinator.com/item?id=20059006


Redis
Neo4j
Cassandra
HBASE
CouchDB
mongoDB
RAVENDB
membase
riak
elasticsearch.
ORACLE
Coherence
Apache
Solr
Software
Engineers

TIMESCALE

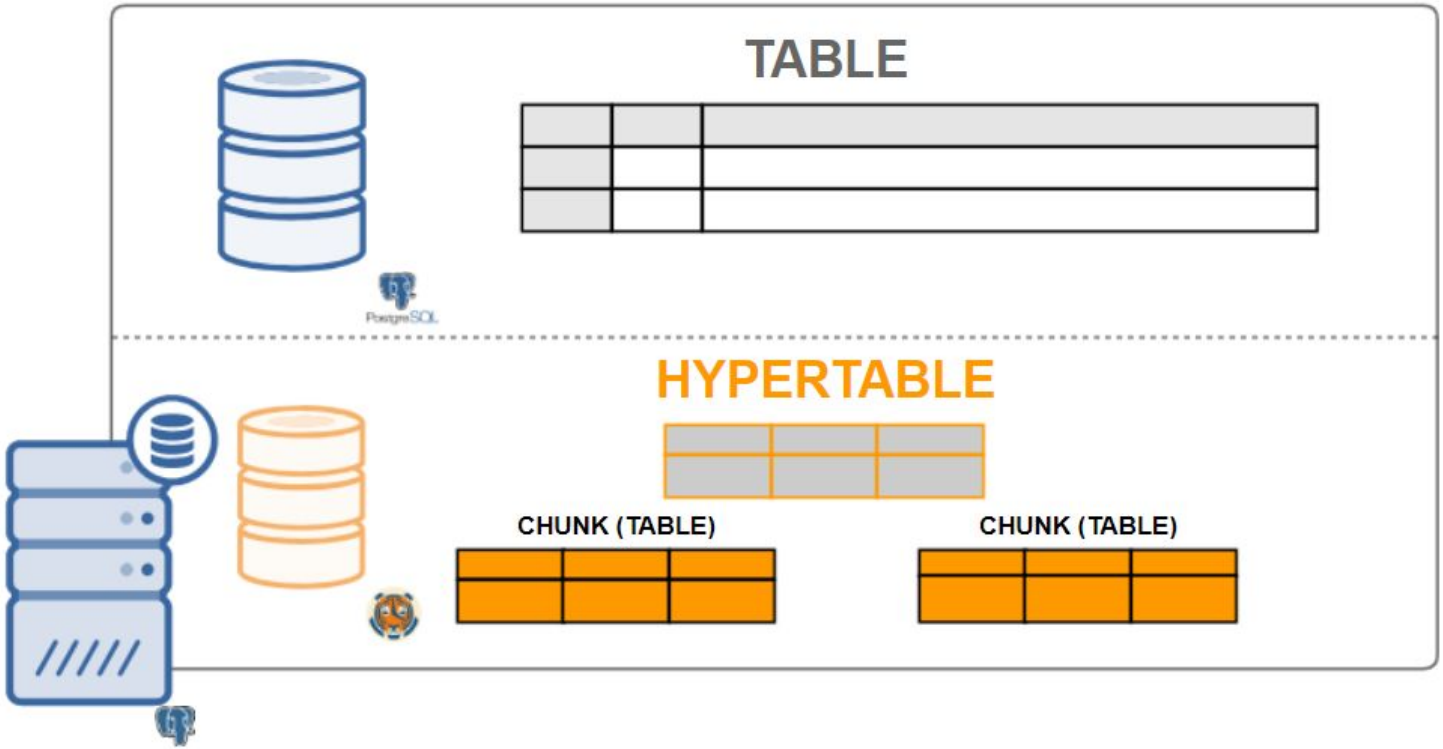
TimescaleDB



- Open source (Apache 2.0, Timescale License)
- Self-hosted / Cloud (AWS, Azure, GCP) / Fully-managed (SaaS)
- PostgreSQL ecosystem + TimescaleDB tools (Promscale, Tune, etc)
- Hypertables and distributed hypertables
- Full SQL + TimescaleDB hyperfunctions
- Real-time continuous aggregates
- Data retention
- Downsampling
- User-defined actions
- Native compression
- Massive scaling
- Function pipelines*

Single-Node Instance

Single-Node Instance

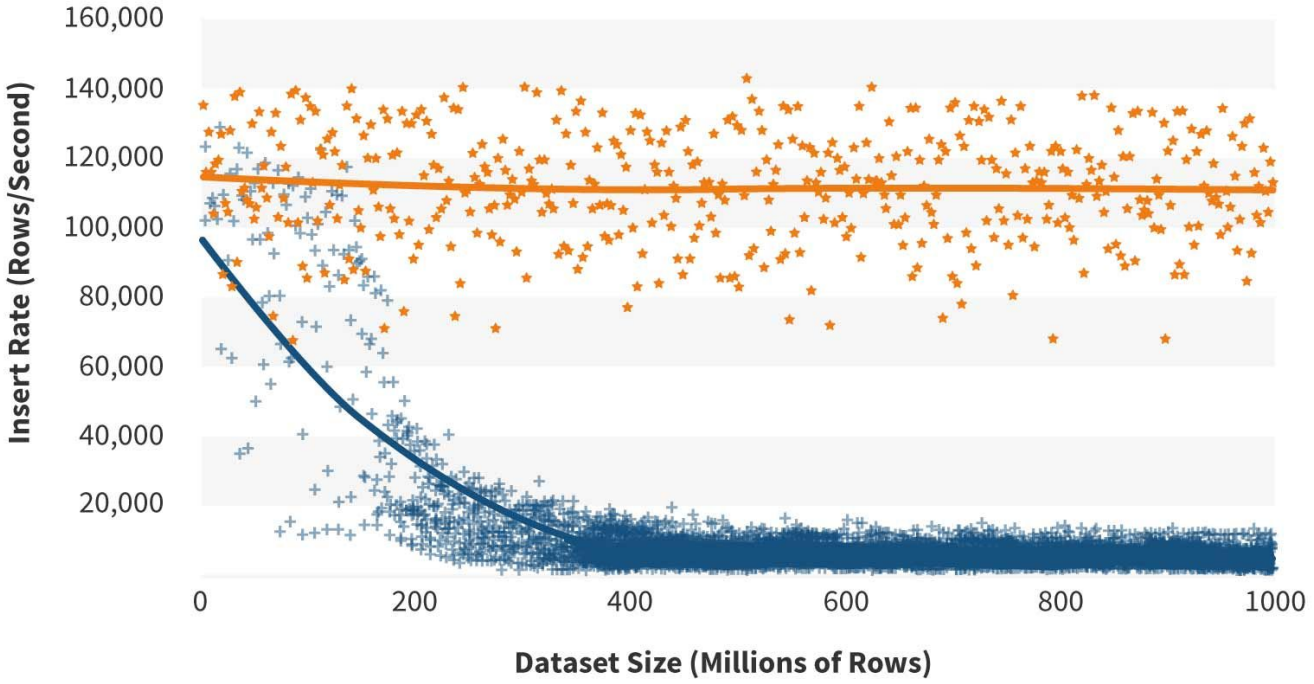


Hypertable of chunks

- Автоматическое секционирование/партиционирование по времени
- Уменьшение размеров секций/партиций (чанков) и индексов к ним
- Увеличение скорости вставки
- Упрощение обслуживания и счастье для DBA ❤️

```sql
CREATE TABLE telemetries (
    imei        TEXT                NOT NULL,
    time        TIMESTAMPTZ         NOT NULL,
    latitude    DOUBLE PRECISION    NOT NULL,
    longitude   DOUBLE PRECISION    NOT NULL,
    speed       SMALLINT            NOT NULL,
    course      SMALLINT            NOT NULL,
    CONSTRAINT telemetries_pkey PRIMARY KEY (imei, time)
);

SELECT * FROM create_hypertable (
    'telemetries', 'time',
    chunk_time_interval => INTERVAL '7 days'
);

INSERT INTO telemetries VALUES ...;

SELECT
    time_bucket('30 days', time) AS bucket,
    imei,
    avg(speed) AS avg,
    max(speed) AS max
FROM telemetries
WHERE speed > 0
GROUP BY imei, bucket
ORDER BY imei, bucket;
```

Ingest Rate: PostgreSQL vs. Timescale

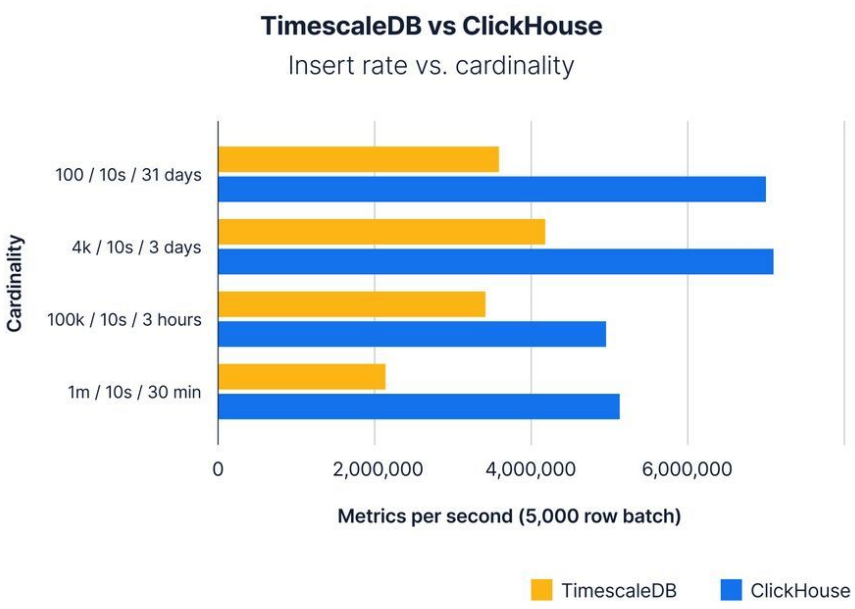


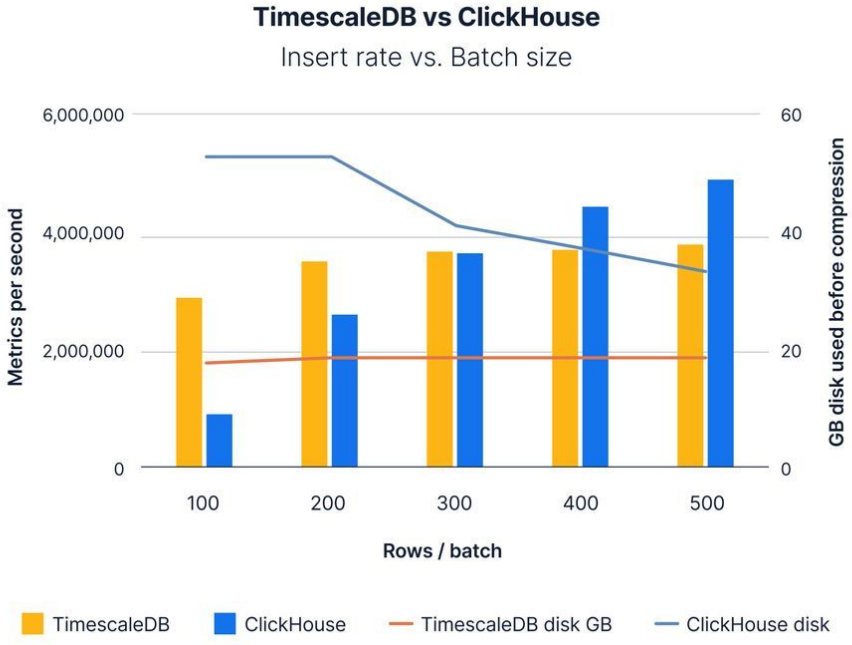
PostgreSQL
TimescaleDB

Insert batch size: 10,000 Rows
Cache: 16 GB Memory

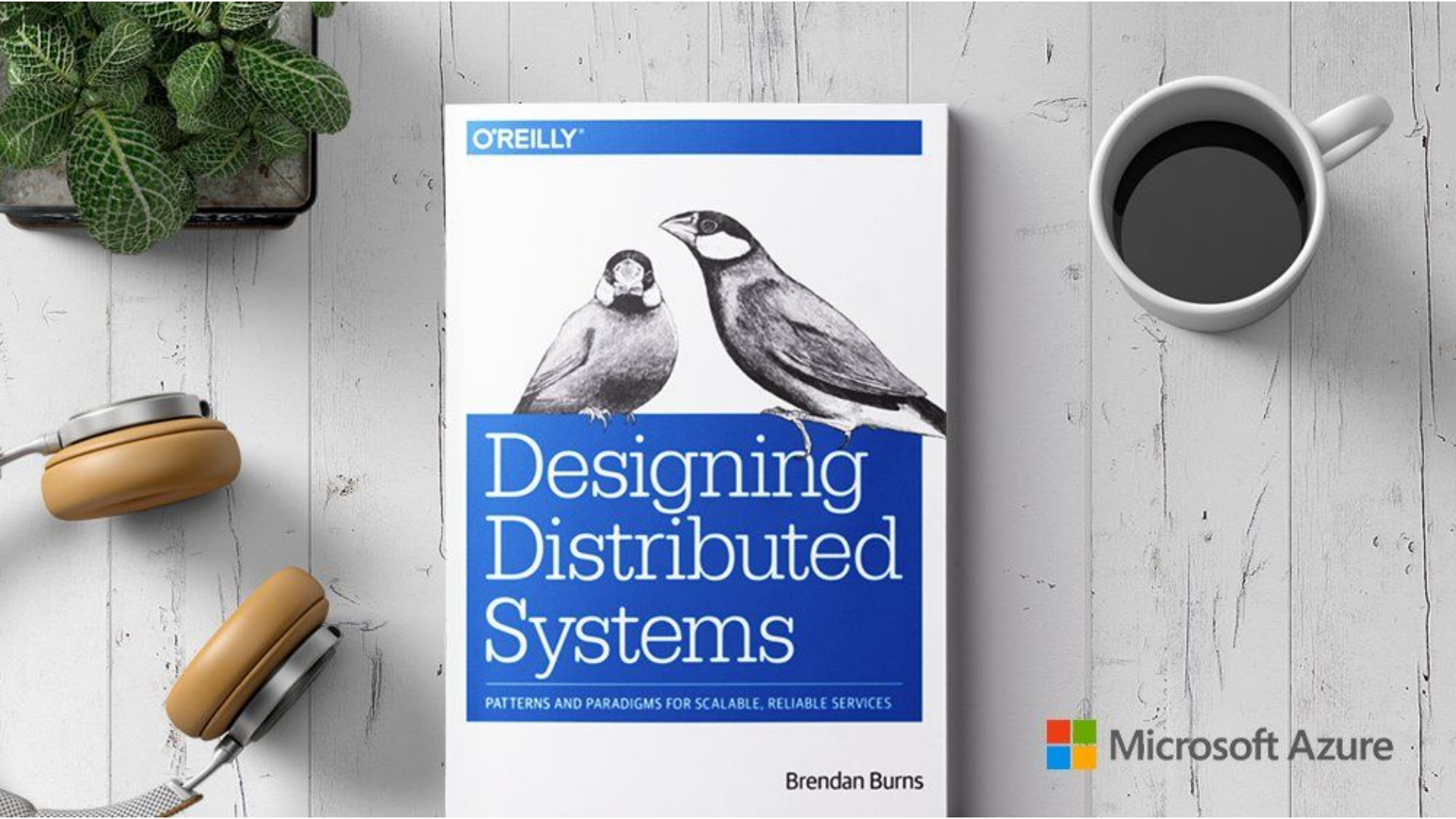
Final avg. throughput: 5k (PG) vs. 111k (TS)
Time to load 1B rows: 37.9h (PG) vs. 2.6h (TS)

Benchmarking, not “Benchmarketing”





Multi-Node Cluster


O'REILLY
Designing Distributed Systems
PATTERNS AND PARADIGMS FOR SCALABLE, RELIABLE SERVICES
Brendan Burns
Microsoft Azure

Clustering system for horizontal scaling

MySQL -> Vitess (Cloud Native Computing Foundation graduated)

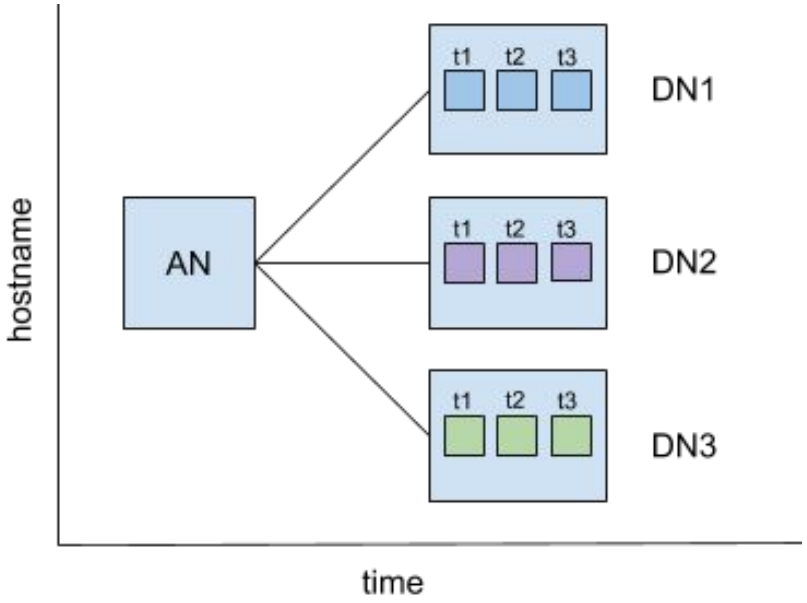
PostgreSQL -> ?

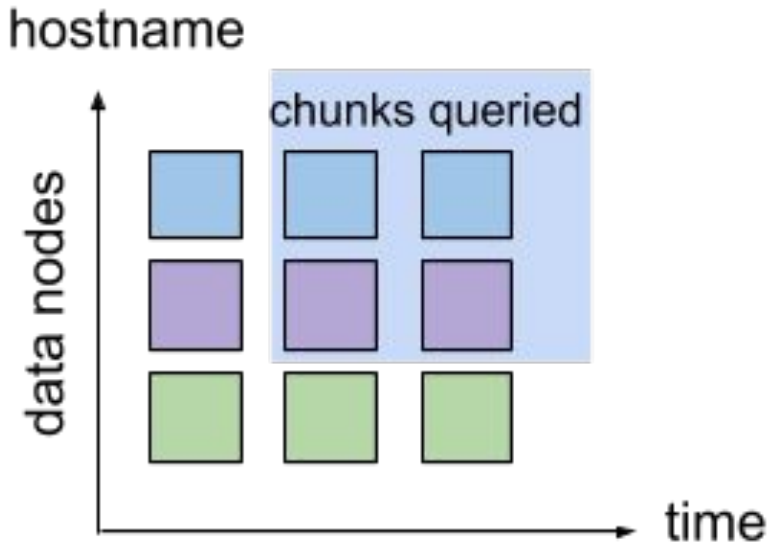
Distributed hypertables

- Time-and-space шардирование без необходимости ребалансировки
- Узел доступа (Access Node) и множество узлов данных (Data Nodes)
- Распараллеливание запросов к DN и агрегация результатов на AN
- Нативная репликация (replication_factor)

Multi-Node Cluster

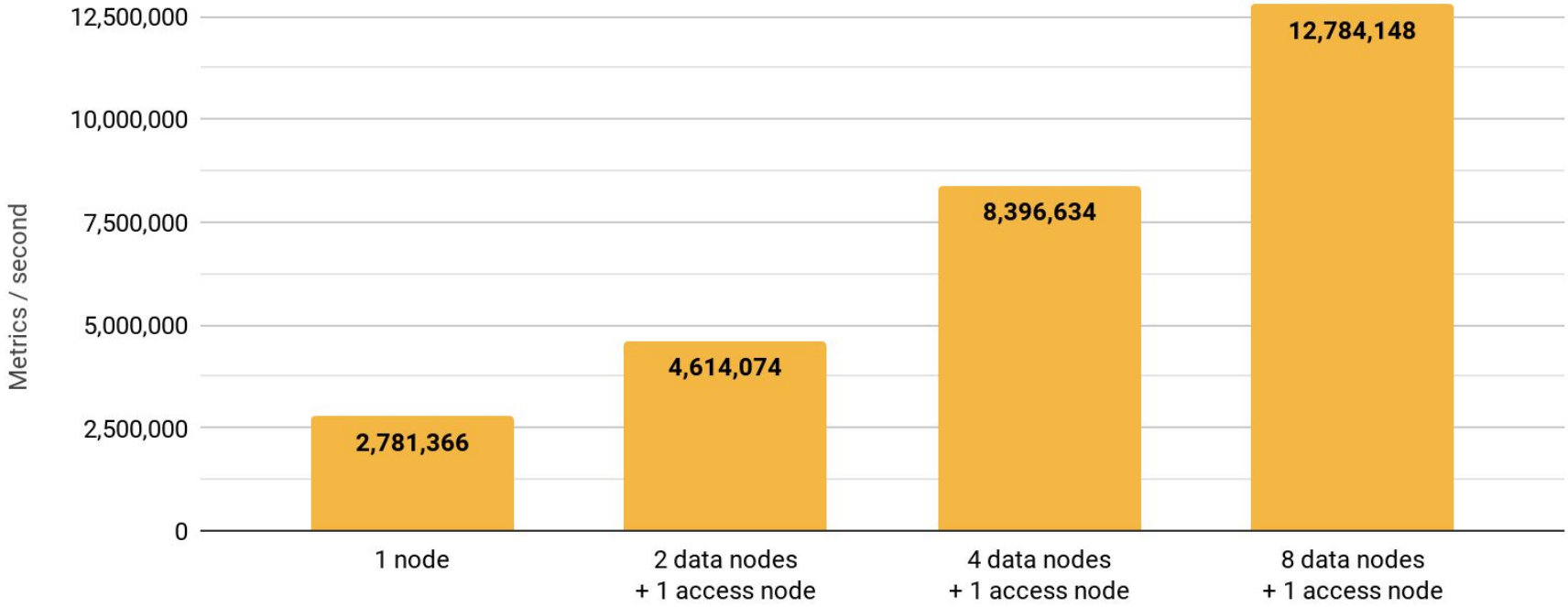
AN (Access Node) - узел доступа
DN (Data Node) - узел данных



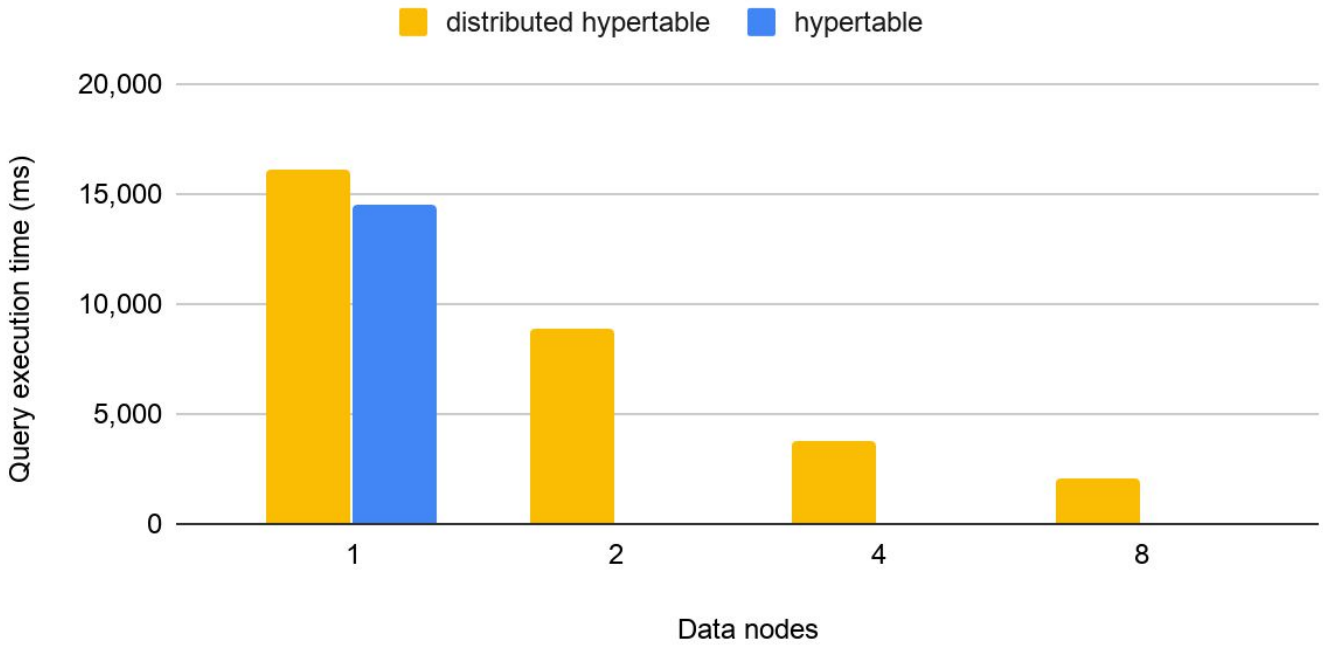


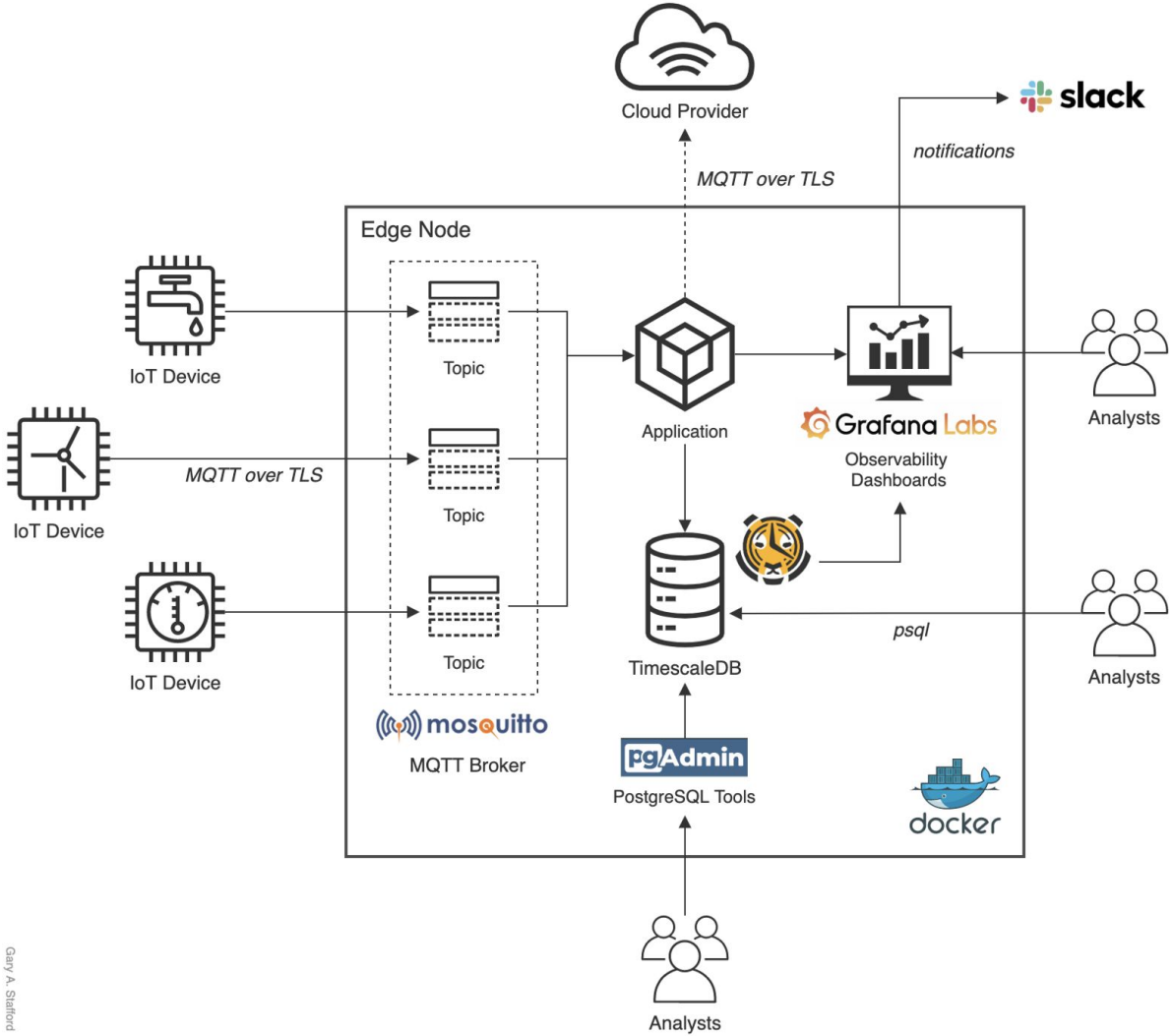
```sql
CREATE TABLE telemetries (
    imei        TEXT                NOT NULL,
    time        TIMESTAMPTZ         NOT NULL,
    latitude    DOUBLE PRECISION    NOT NULL,
    longitude   DOUBLE PRECISION    NOT NULL,
    speed       SMALLINT            NOT NULL,
    course      SMALLINT            NOT NULL,
    CONSTRAINT telemetries_pkey PRIMARY KEY (imei, time)
);

SELECT * FROM add_data_node('data_node_1', host => 'pg_data_node_1');
SELECT * FROM add_data_node('data_node_2', host => 'pg_data_node_2');
SELECT * FROM add_data_node('data_node_3', host => 'pg_data_node_3');

SELECT * FROM create_distributed_hypertable(
    'telemetries', 'time', 'imei',
    chunk_time_interval => INTERVAL '7 days'
);

INSERT INTO telemetries VALUES ...;

SELECT
    time_bucket('30 days', time) AS bucket,
    imei,
    avg(speed) AS avg, max(speed) AS max
FROM telemetries
WHERE speed > 0
GROUP BY imei, bucket
ORDER BY imei, bucket;
```

Insert performance as cluster size increases



Hourly average CPU usage per host over the past 24 hours (full aggregation)





Limitations

Hypertable limitations:

- Foreign key constraints referencing a hypertable are not supported.
- Time dimensions (columns) used for partitioning cannot have NULL values.
- Unique indexes must include all columns that are partitioning dimensions.
- UPDATE statements that move values between partitions (chunks) are not supported.

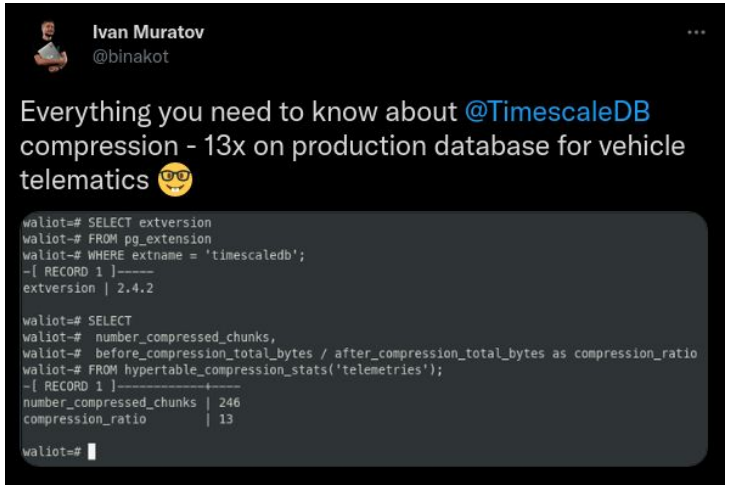
Distributed hypertable limitations:

- Distributed scheduling of background jobs is not supported.
- Continuous aggregates are not supported.
- Joins on data nodes are not supported.
- Tables referenced by foreign key constraints in a distributed hypertable must be present on the access node and all data nodes.

https://docs.timescale.com/timescaledb/latest/overview/limitations

ПРОФИТ?

- Настоящий SQL
- Экосистема PostgreSQL
- Масштабирование
- Открытый исходный код
- Свободное использование
- Огромное и дружелюбное комьюнити



Function pipelines: Building functional programming into PostgreSQL using custom operators

Grafana ❤️ TimescaleDB
Zabbix ❤️ TimescaleDB

Полезняшки

@binakot

https://krd.dev

https://t.me/krddevdays

https://www.timescale.com

https://blog.timescale.com

https://youtu.be/vbJCq9PhSR0

https://youtu.be/1C2VGD90KGk

https://github.com/binakot/Multi-Node-TimescaleDB

https://hub.docker.com/r/binakot/postgresql-postgis-timescaledb

Спасибо за внимание!

https://github.com/binakot/Multi-Node-TimescaleDB