

About

~2yrs at Crate.io

DevRel/Field Engineering/Support/ Integrations/...

Offices

San Francisco, Berlin, Dornbirn (AT)

Talk to me about

Rust, Raspberry Pis, Tech!



Agenda

About machine data

Why is it special?

CrateDB fundamentals

A deep-ish dive

Labs: Connecting sensors to CrateDB

Three parts: Python, CrateDB, Grafana

Wrap up

Next steps, more webinars!



Requirements

Python

Application code

CrateDB (or use ours)

To store the data. Use ours or a Docker image

Grafana

Hosted Grafana @ Grafana.com

A place to run all this

Your computer

Your chance to start preparing:)



git clone

Find all files here

https://github.com/crate/webinar.101

Find a cluster here

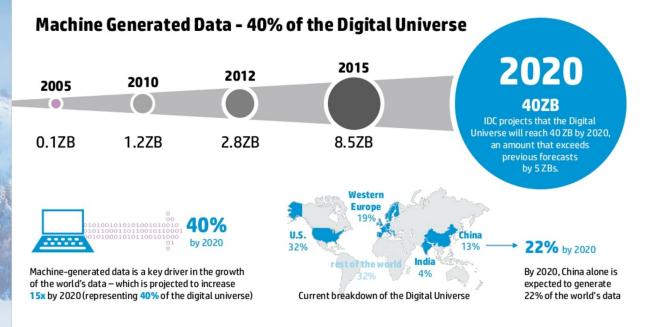
http://147.75.64.195:4280





Machine Data





Source: HPE Jun 2016

http://www.slideshare.net/penumuru/harness-the-power-of-big-data-with-oracle-63438438/9

Machine Data Characteristics

Millions of data points/second

Streaming in from sensors, devices, logs, etc.

Data diversity

Structured & unstructured JSON, Blobs

Real-time query performance

Monitoring & alerting

Complex queries of big data volumes

With Terabytes of historic data

Growth

Adding sources often means exponential growth



Machine Data

Internet of Things

Sensors, cameras, ...

Wearables, Gadgets

Location data, interaction data, ...

Logs & Monitoring data

Component health monitoring, access logs, ...

Industry 4.0, Digitization

Production line insights, automation, ...

Vehicles

Location data, health data, ...



Clickdrive.io

Fleet management & vehicle tracking
Vehicle health and tracking data

High ingest rate 2,000 data points per car, per second

In-depth & real-time analysis

Predictive maintenance, accident
reconstruction, route/driver efficiency



Roomonitor

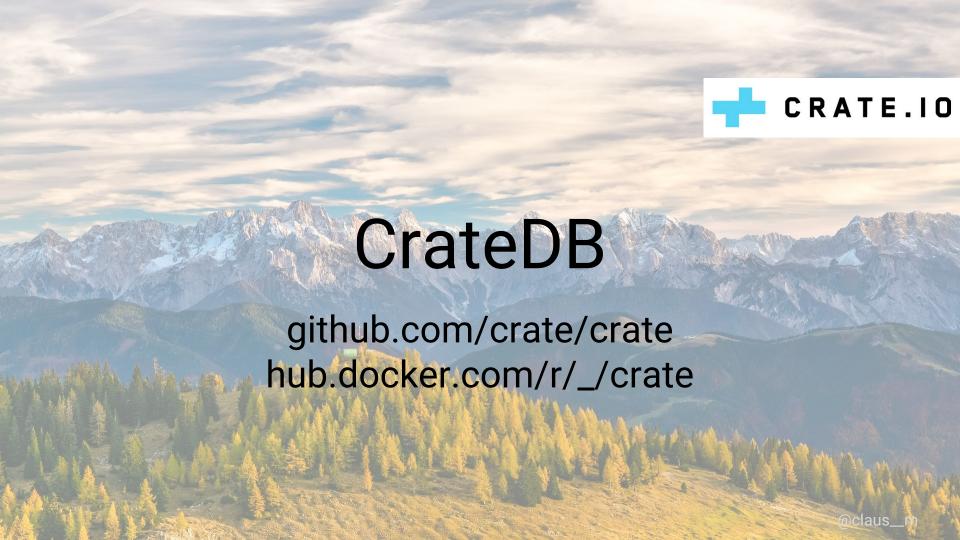
Smart apartments

Monitoring & control climate, occupancy, noise, access

Better efficiency, safer environment

Alerts: AC/heating on with window open, noisy neighbors, ...





CrateDB

Shared nothing

All nodes are equal

Partitioning, auto-sharding & replication

Transparent to the user

Multi model: Structured &

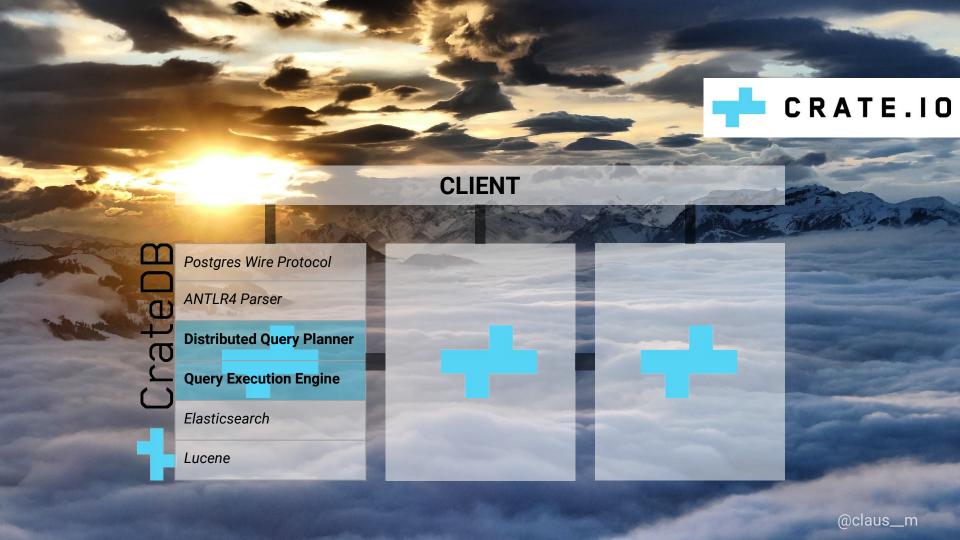
unstructured

Search, queries, aggregates, joins

SQL

SQL!





CrateDB Fundamentals

Disk-based index with in-memory caching
Fast and efficient OS caching

Shards: Units of data
Concurrency by distributing
shards

Distributed query execution engine

"Push down" queries



Lucene: CrateDB Shards

Documents

Rows with expansible columns

Fulltext search: Inverted index

Analyse, tokenise, and search

Compression

LZ4 compression of fields

Field cache

Columnar storage

Data types

Java types: long, int, string, ...



Clustering: Shard Management

On-disk storage

Multiple files

Replication

Copies of initial files (primaries)

Distribution

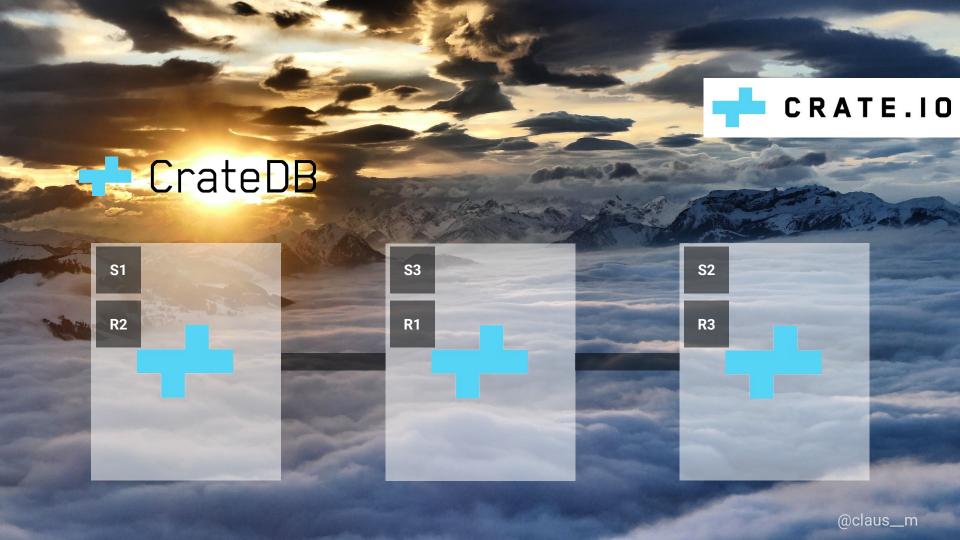
Shuffle around shards (primaries & replicas)

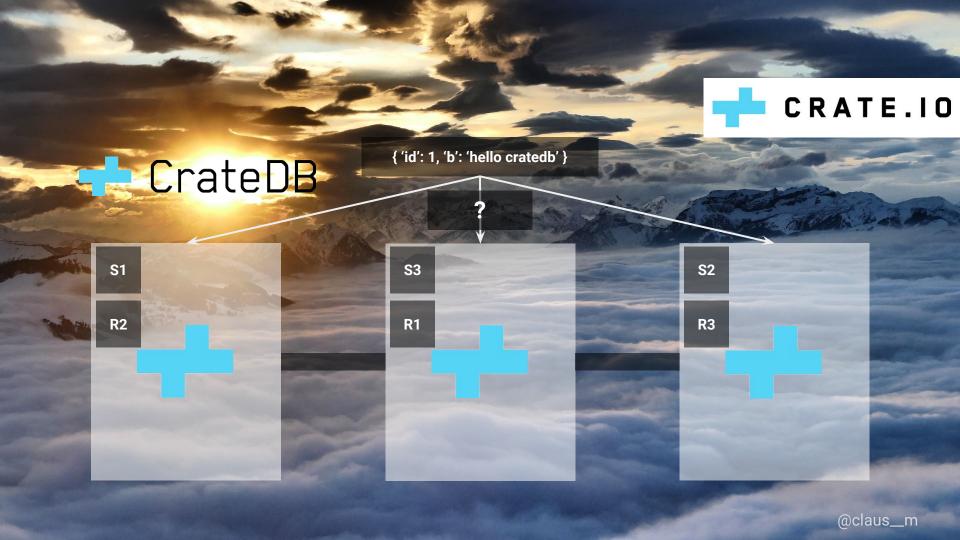
Cluster state

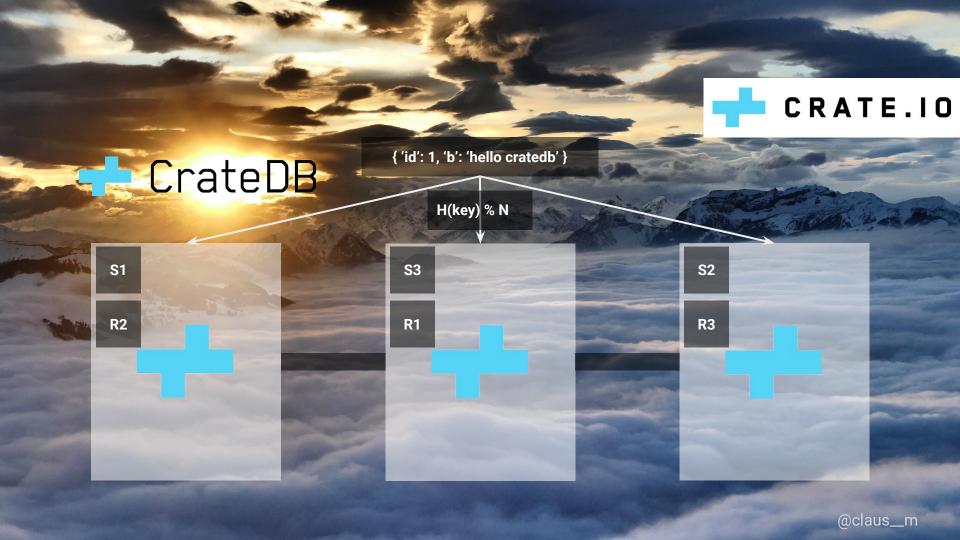
Stores shard locations based on _id

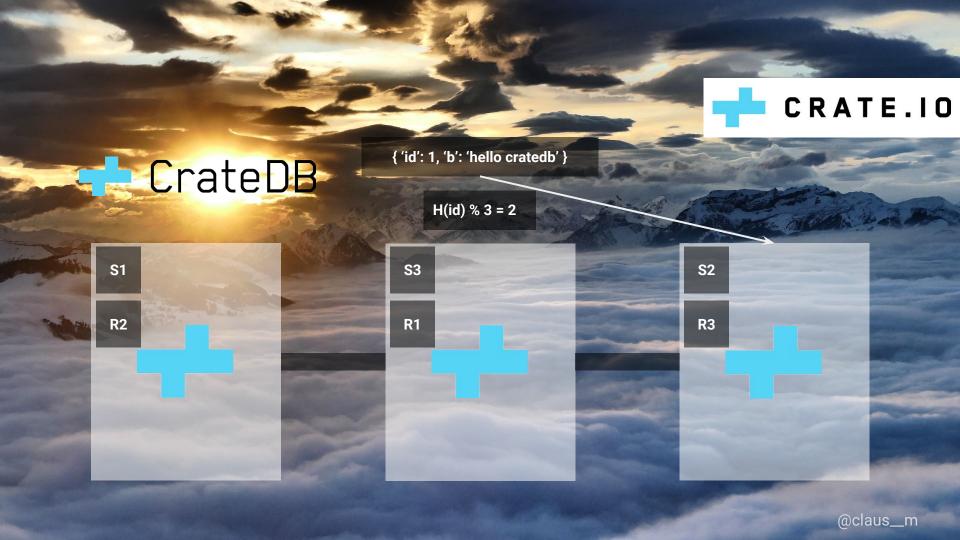






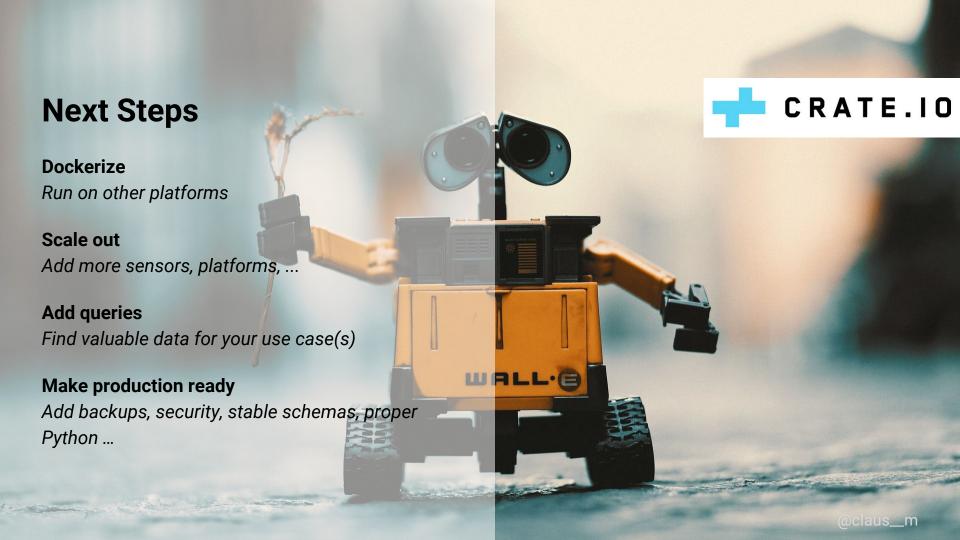


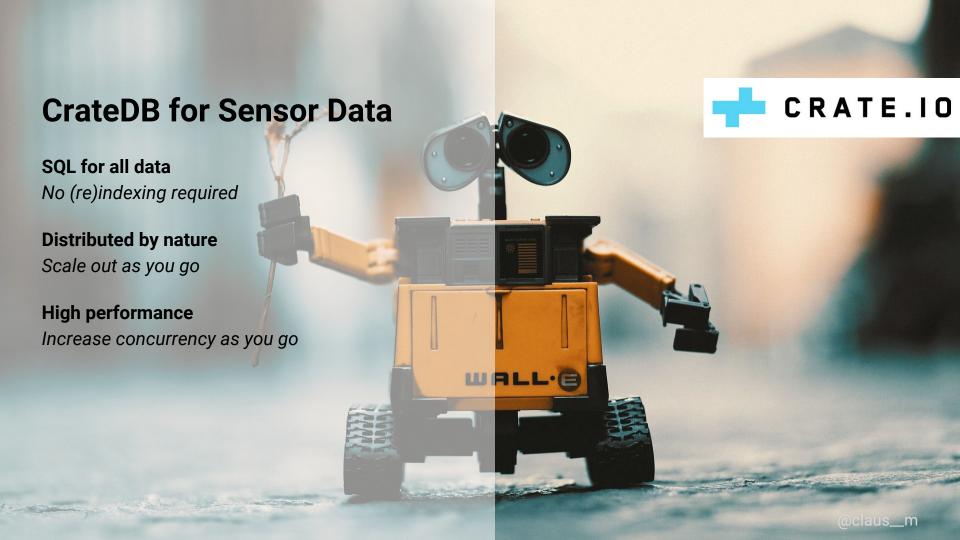












Links

https://github.com/crate

Follow us on twitter

@crateio @claus__m

Next webinar: Log data, 25th April

