Applications of Machine Learning in Distributed Systems

Philipp Wiesner

Adaptive Resource Management (ARM)
in Distributed and Operating Systems (DOS)

Technische Universität Berlin, 27 Jan 2021

Agenda

- AI in Distributed Systems
- Time Series Forecasting
  - Renewable-Aware Scheduling
  - Self-Optimizing Systems
- AIOps
  - Self-Healing Systems
  - Research Results: Anomaly Detection
- Summary

Cloud Computing

Distributed Stream/Batch Processing

Peer-to-Peer Networks (P2P)

Real-Time Systems

Internet of Things (IoT)

Fog & Edge Computing

What are Distributed Systems?

Routing Algorithms

Telecomunication Networks

Wireless Sensor Networks

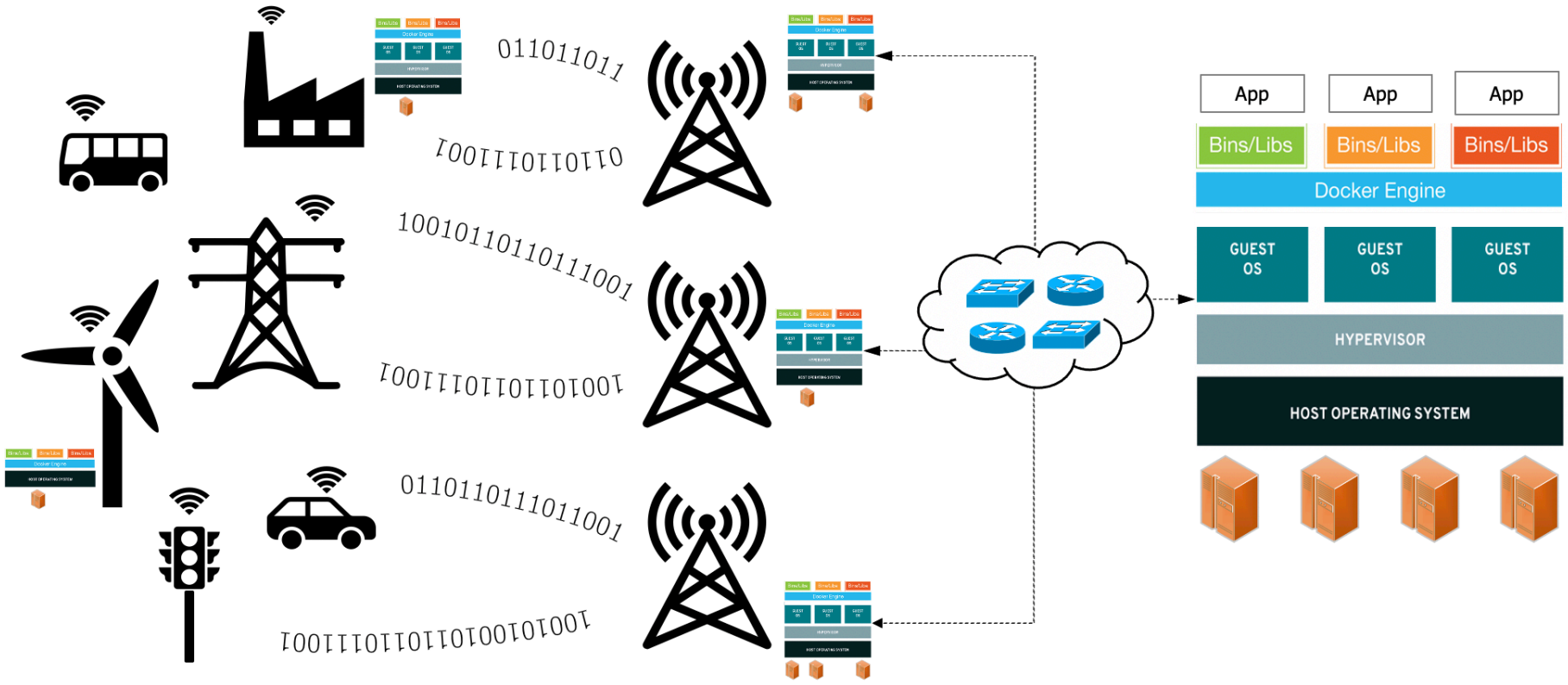
Smart Cities

Cluster Computing

Industry 4.0

Distributed Databases

Complexity in Distributed Systems



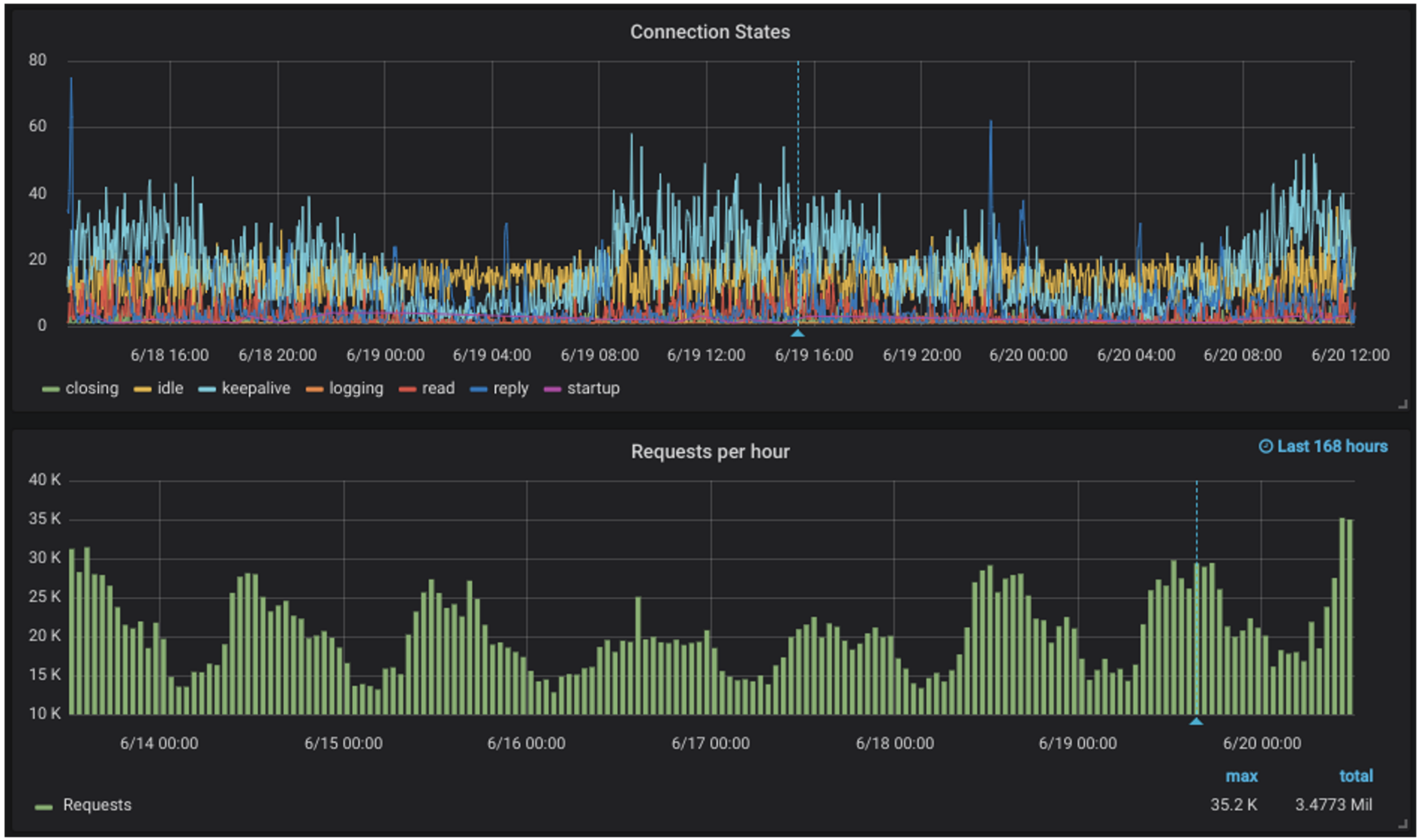
Complexity in Distributed Systems

Big data and fast data are solved!
How to keep the gigantic clusters up and running is the main challenge!

Prof. Michael Stonebraker, MIT Computer Science & AI Lab,
Turing price winner 2015, Statement from September 2018

(that's 26 seconds downtime per month)

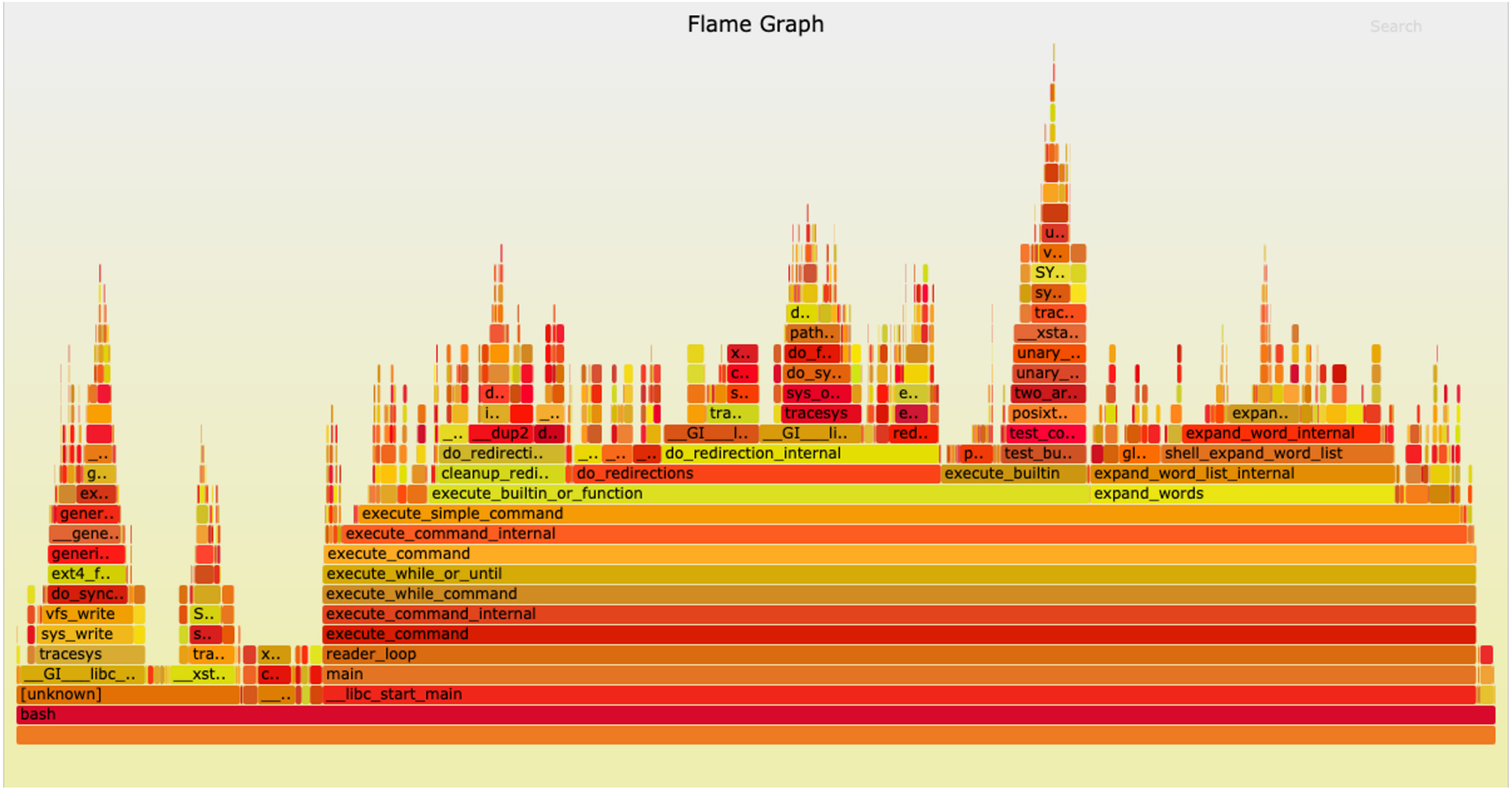
Types of Data: Metrics



Types of Data: Logs / Text Data

```
 9  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /tipps/google-suche.html HTTP/1.1" 200 6735 "https://
10  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /styles/formats.css HTTP/1.1" 200 6831 "http://www.st
11  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /styles/formatp.css HTTP/1.1" 200 261 "http://www.sti
12  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /tipps/such-vorschlaege.png HTTP/1.1" 200 8164 "http:
13  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /styles/kontakt.png HTTP/1.1" 200 898 "http://www.sti
14  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /tipps/suchbegriff-operator.png HTTP/1.1" 200 8260 "h
15  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /tipps/google-autovervollstaendigung.png HTTP/1.1" 20
16  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /styles/englisch-30.png HTTP/1.1" 200 1035 "http://ww
17  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /logo.png HTTP/1.1" 200 6006 "http://www.stichpunkt.d
18  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /favicon.ico HTTP/1.1" 200 3262 "-" "Mozilla/5.0 (Win
19  61.159.121.13 - - [10/Apr/2018:14:08:06 +0200] "GET /favicon.ico HTTP/1.1" 499 0 "-" "Mozilla/5.0 (Windows
20  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:01 +0200] "GET /styles/formatsx.css HTTP/1.1"
21  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:26 +0200] "GET /styles/background-header.png
22  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:26 +0200] "GET /styles/fonts/OpenSans-Regular
23  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:27 +0200] "GET /abbreviations/internet-acrony
24  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:27 +0200] "GET /abbreviations/text-message.pn
25  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:27 +0200] "GET /styles/contact.png HTTP/1.1"
26  2003:6f:b121:6e57:3014:770e:2d1:131d - - [10/Apr/2018:14:08:27 +0200] "GET /styles/formatp.css HTTP/1.1"
27  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/style.css?ver=4.9.3 HTTP/1.
28  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman-child/style.css?ver=4.9.3 H
29  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/assets/front/css/main.min.c
30  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman-child/style.css?ver=3.3.25
31  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/assets/front/css/font-aweso
32  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-includes/js/jquery/jquery.js?ver=1.12.4 HTTP/1.1"
33  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-includes/js/jquery/jquery-migrate.min.js?ver=1.4.
34  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman-child/custom.js?ver=1 HTTP/
35  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/plugins/hueman-addons/addons/assets/front
36  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-includes/js/underscore.min.js?ver=1.8.3 HTTP/1.1"
37  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-includes/js/wp-embed.min.js?ver=4.9.3 HTTP/1.1" 2
38  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/assets/front/js/scripts.min
39  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-includes/js/wp-emoji-release.min.js?ver=4.9.3 HTT
40  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/assets/front/img/sidebar/s-
41  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /wp-content/themes/hueman/assets/front/img/sidebar/s-
42  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /zeichen/vuvuzela-einzelne-note.mp3 HTTP/1.1" 304 0 "
43  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /zeichen/vuvuzela-harmonische-toene-1-bis-3.mp3 HTTP/
44  63.172.164.99 - - [10/Apr/2018:17:44:54 +0200] "GET /zeichen/vuvuzela-fussball-stadium.mp3 HTTP/1.1" 206
```

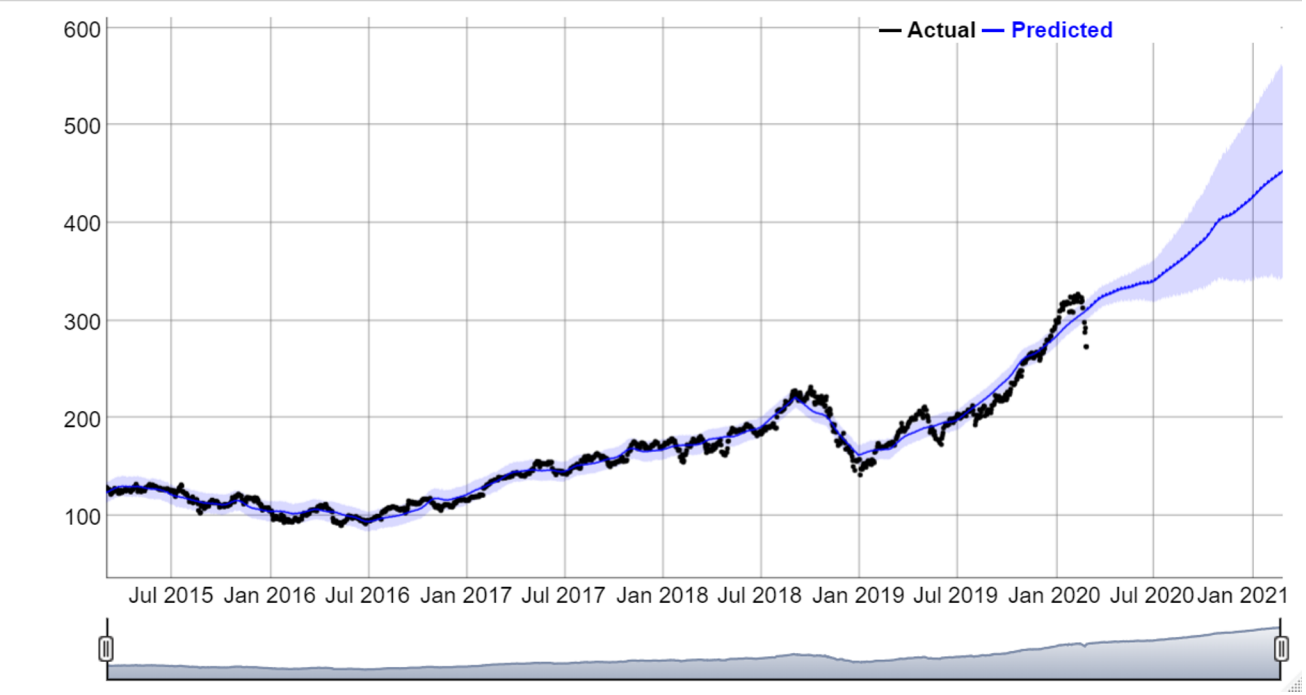
Types of Data: Traces



Time Series Forecasting

Learning from the past...

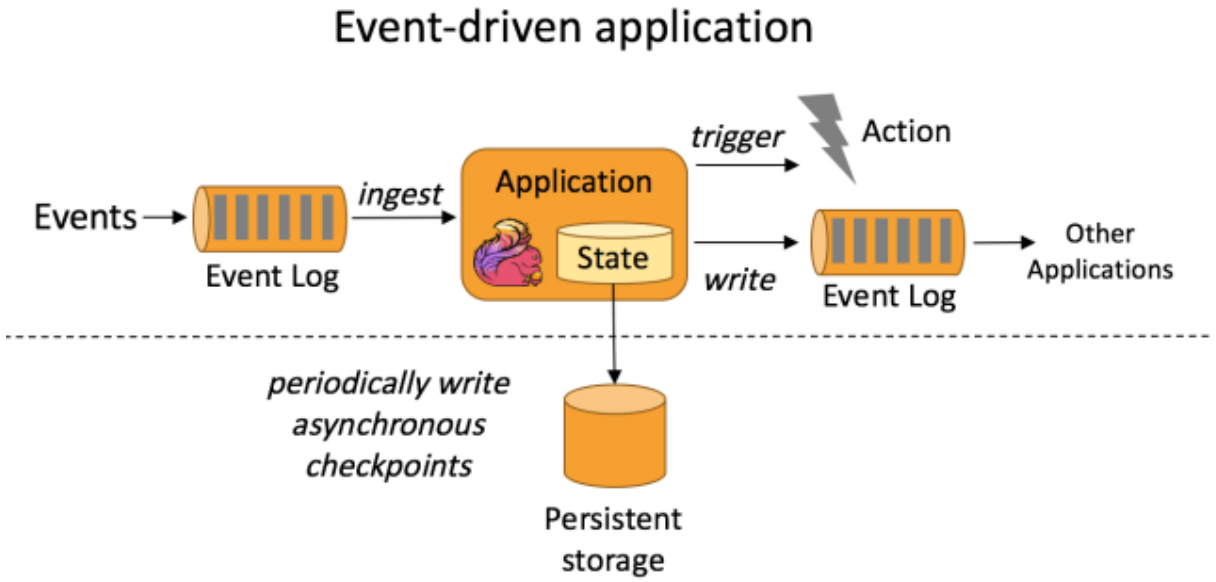
… to predict the future



https://levelup.gitconnected.com/time-series-forecasting-with-prophet-in-r-a9ee81dc82e1

Automatic System Configuration

- Distributed Stream Processing, e.g. filtering Tweets



https://flink.apache.org/usecases.html

Automatic System Configuration

Fault tolerance is expensive:

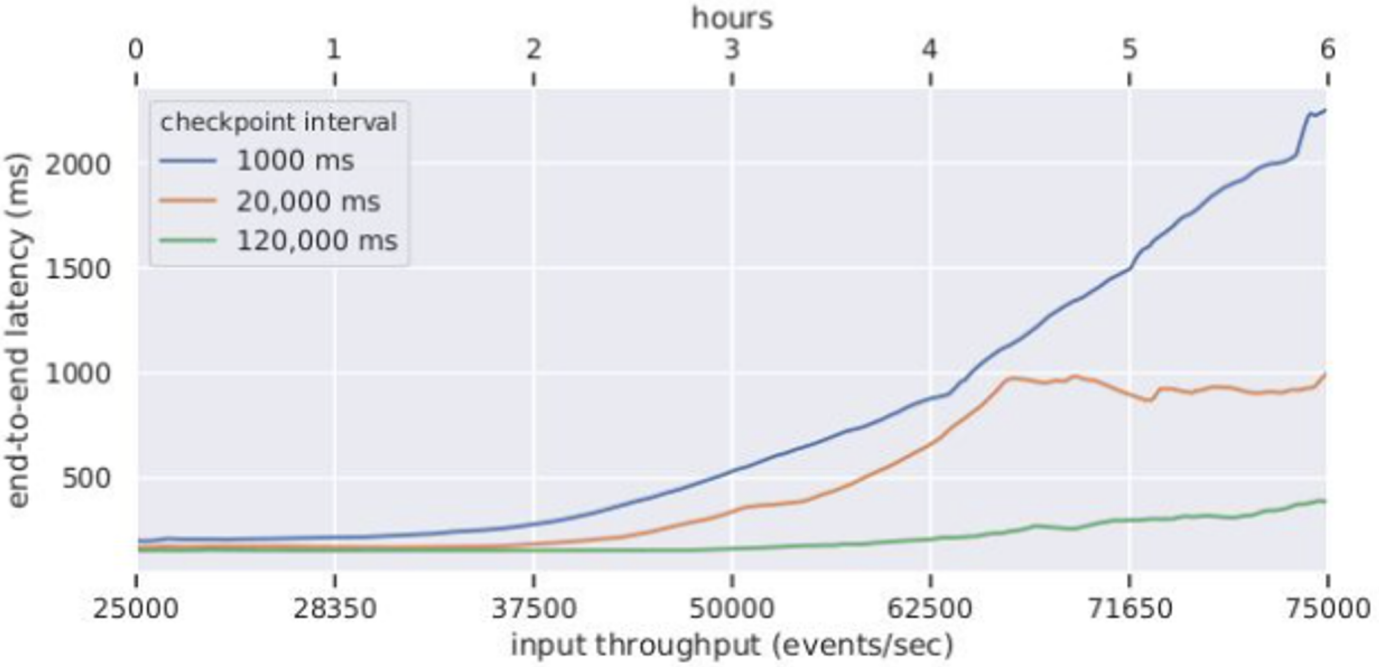
- Every checkpoint write costs performance
- Users can freely configure the *Checkpoint Interval*
- But how?

Manual configuration of jobs is time consuming

- Given the max time a system can be unavailable, how can we find the best performance?

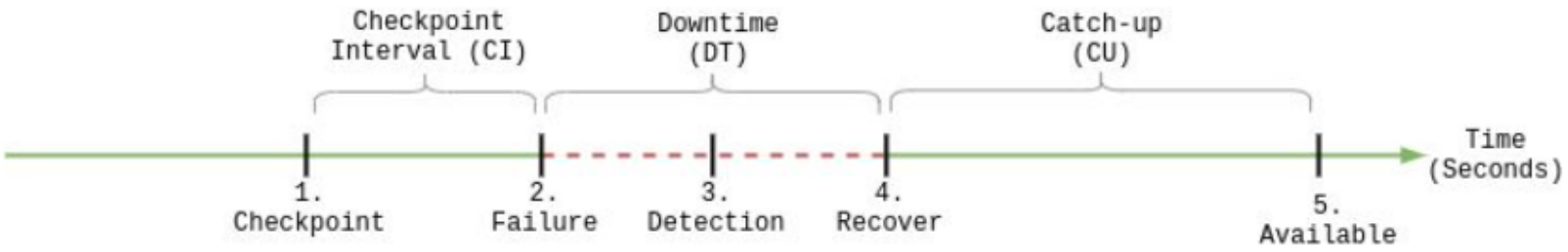
Automatic System Configuration

Configuring streaming jobs: How often should I save a checkpoint?

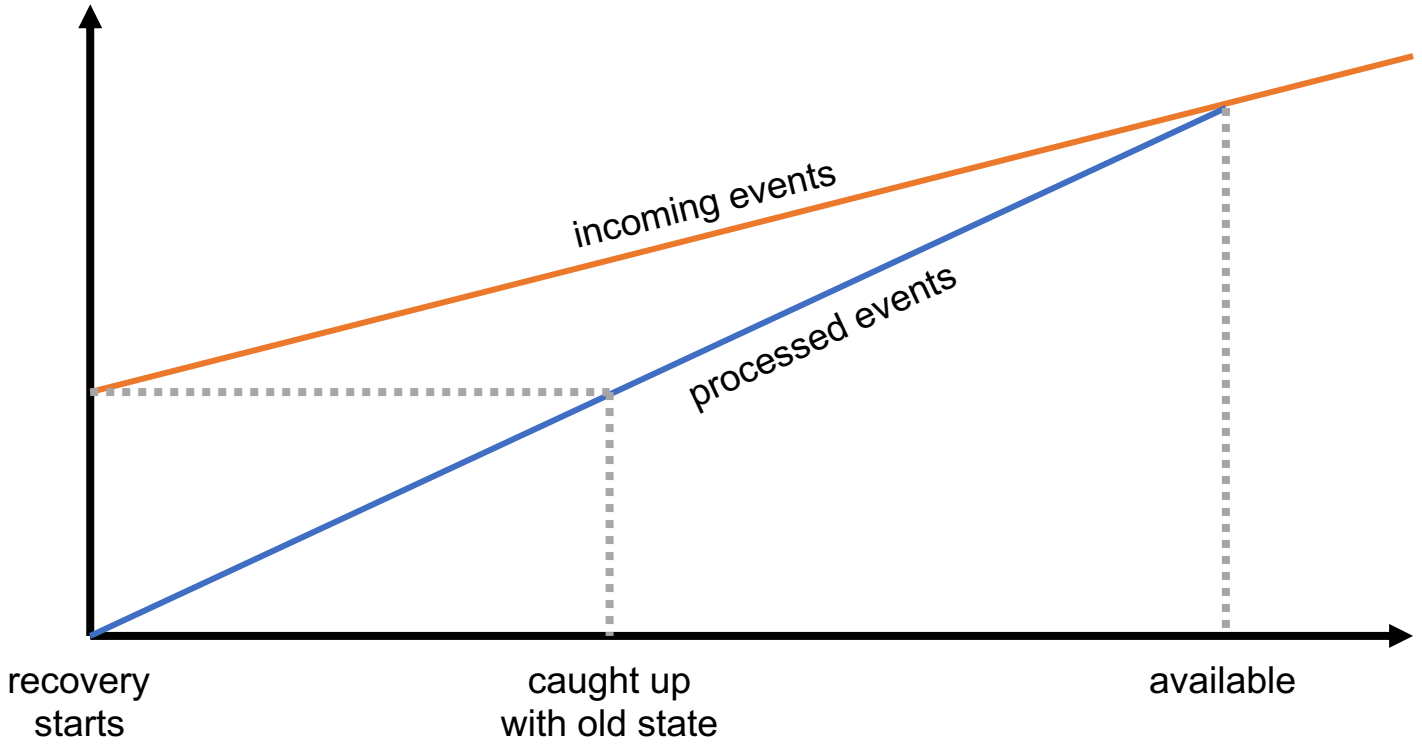


Automatic System Configuration

Total Recovery Time: Time required to catch-up to the latest offset of the incoming event stream from the point at which the failure occurred



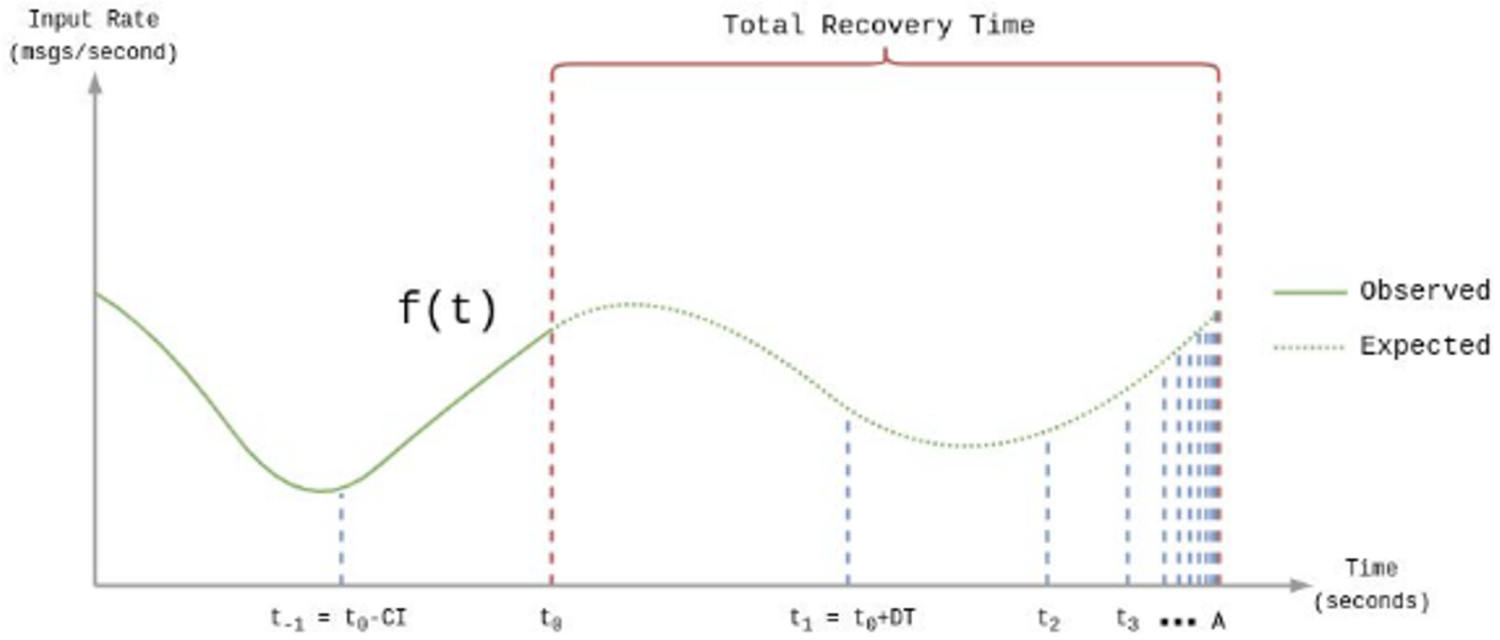
Simplified Catch Up



Problem: Varying workloads

Load Prediction

Model the catch-up as a decreasing geometric sequence



Wrap Up: Automatic System Configuration

Automatically adapt configuration parameters for a distributed application without expert knowledge.

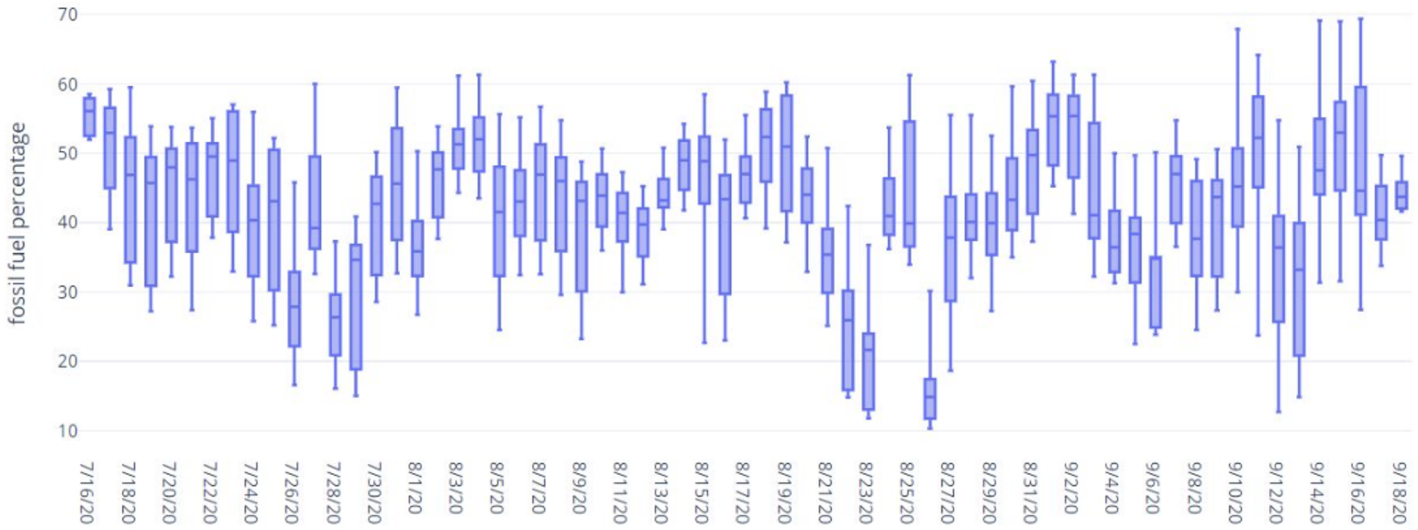
Checkpoint interval example:

- High load expected: Hard to catch up
- Low load expected: Easy to catch up

Renewable-Aware Scheduling

Carbon Intensity: Emissions per energy (gCO2eq/kWh)

Fossil Fuel Percentage: Fraction of energy generated via fossil fuels

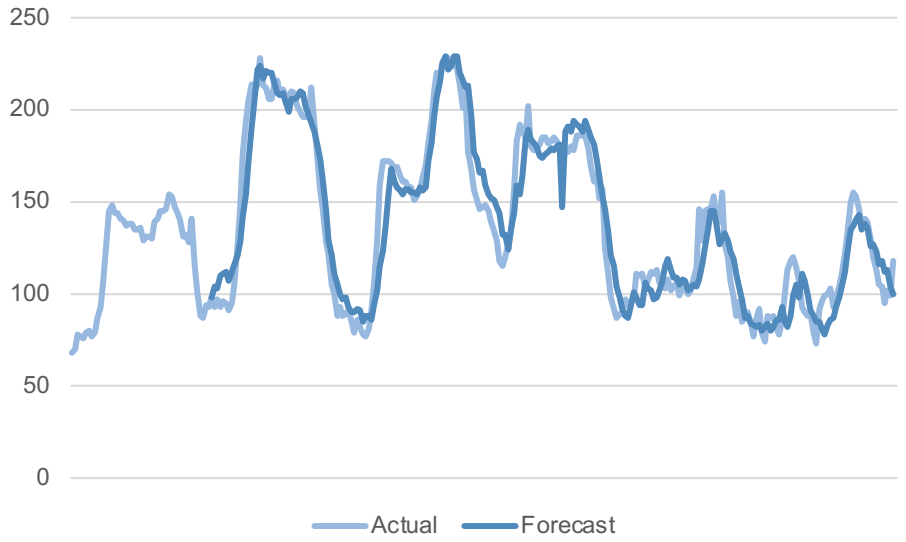


"Dirtyness" of the energy mix varies over time

Renewable-Aware Scheduling

Forecast carbon intensity

- Forecast energy production per production type and determine carbon intensity [1]
  - Wind onshore: 11 gCO2eq/kWh
  - Nuclear: 12 gCO2eq/kWh
  - Solar: 41 gCO2eq/kWh
  - Gas: 490 gCO2eq/kWh
  - Coal: 820 gCO2eq/kWh
- Forecast cross-border flows
- Forecast demand



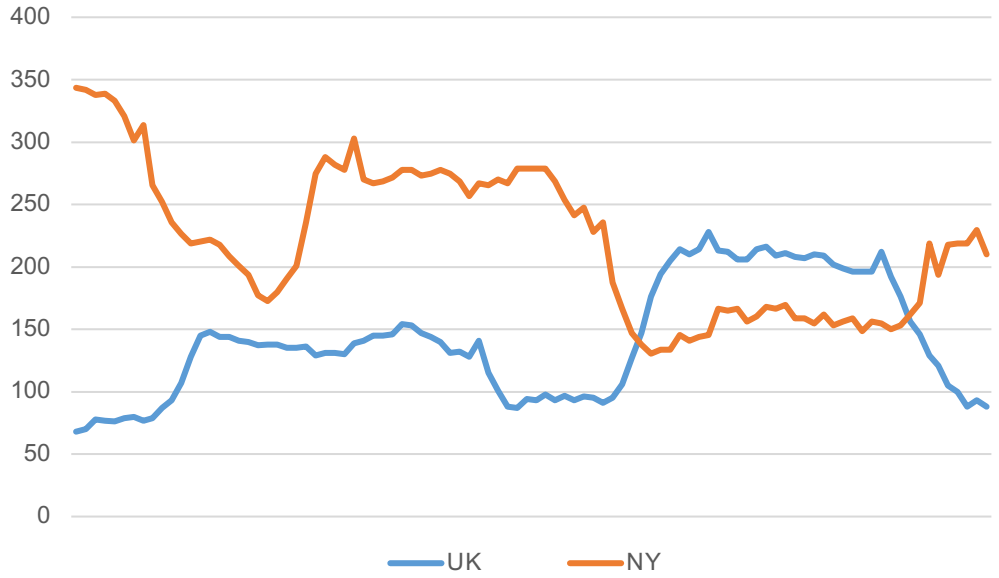
Enhance prediction with data from the “future”

- Weather forecasts
- Demand reservations
- Production Inertia

[1] Bruckner et al. *Annex III of IPCC Special Report on Renewable Energy Sources and Climate Change Mitigation: Recent Renewable Energy Cost and Performance Parameters,* 2014

Renewable-Aware Scheduling

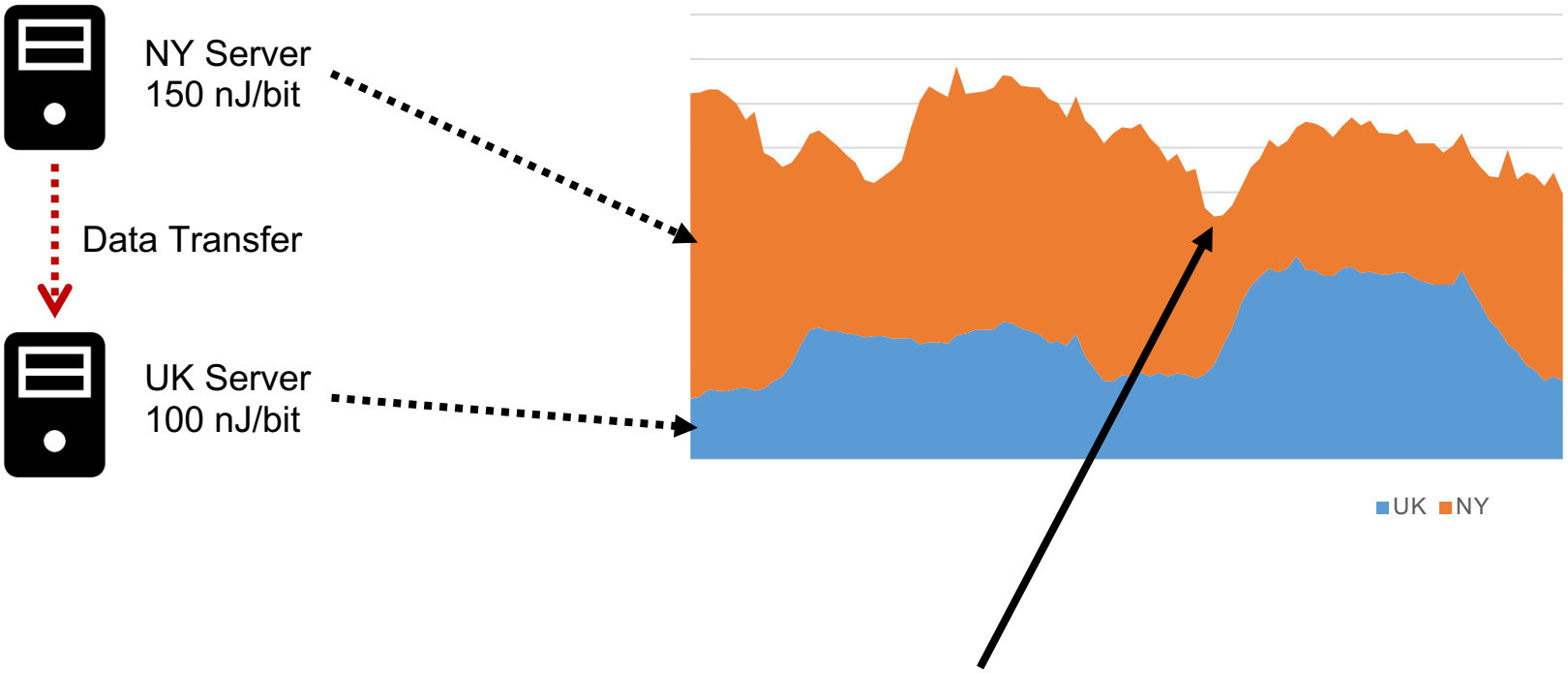
Carbon intensity prediction over the course of two days in the UK and NY



Due to different timezones “clean” periods do not overlap

-> When is the best moment to transfer a large bulk of data?

Renewable-Aware Scheduling



Find **point in time / interval**

where the energy is the cleanest

AIOps

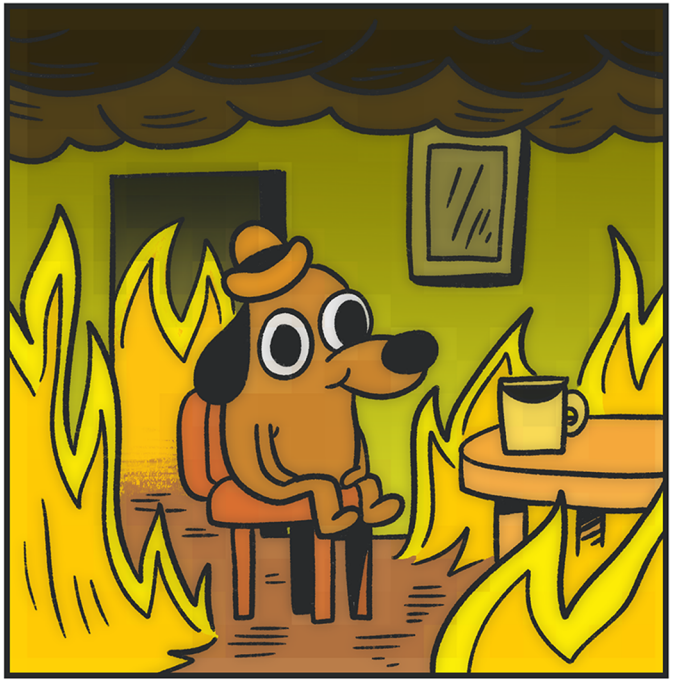
Why AIOps?

Humans don’t scale well

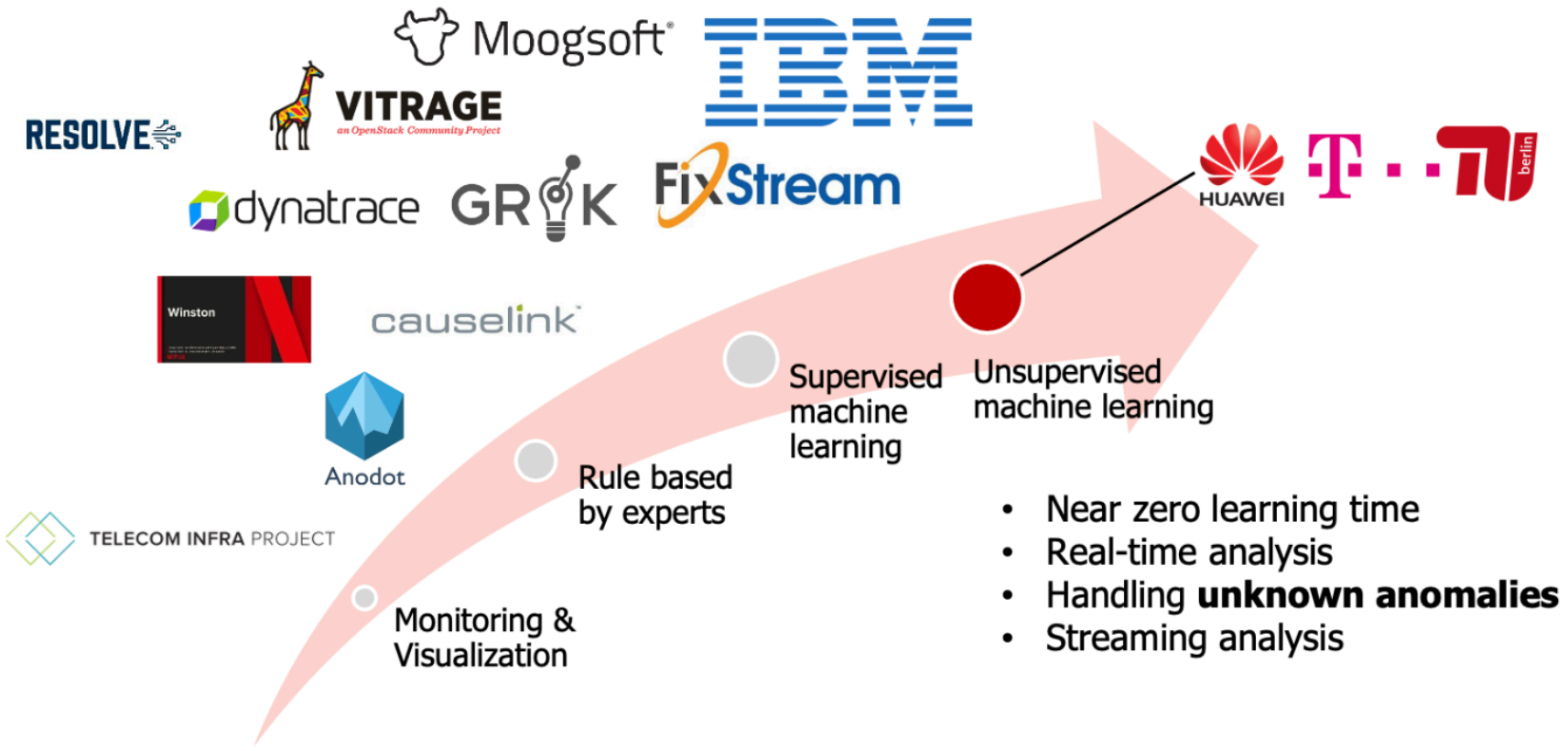
- 5 servers → 1 operator
- 5000 servers → 1000 operators?

In practice:

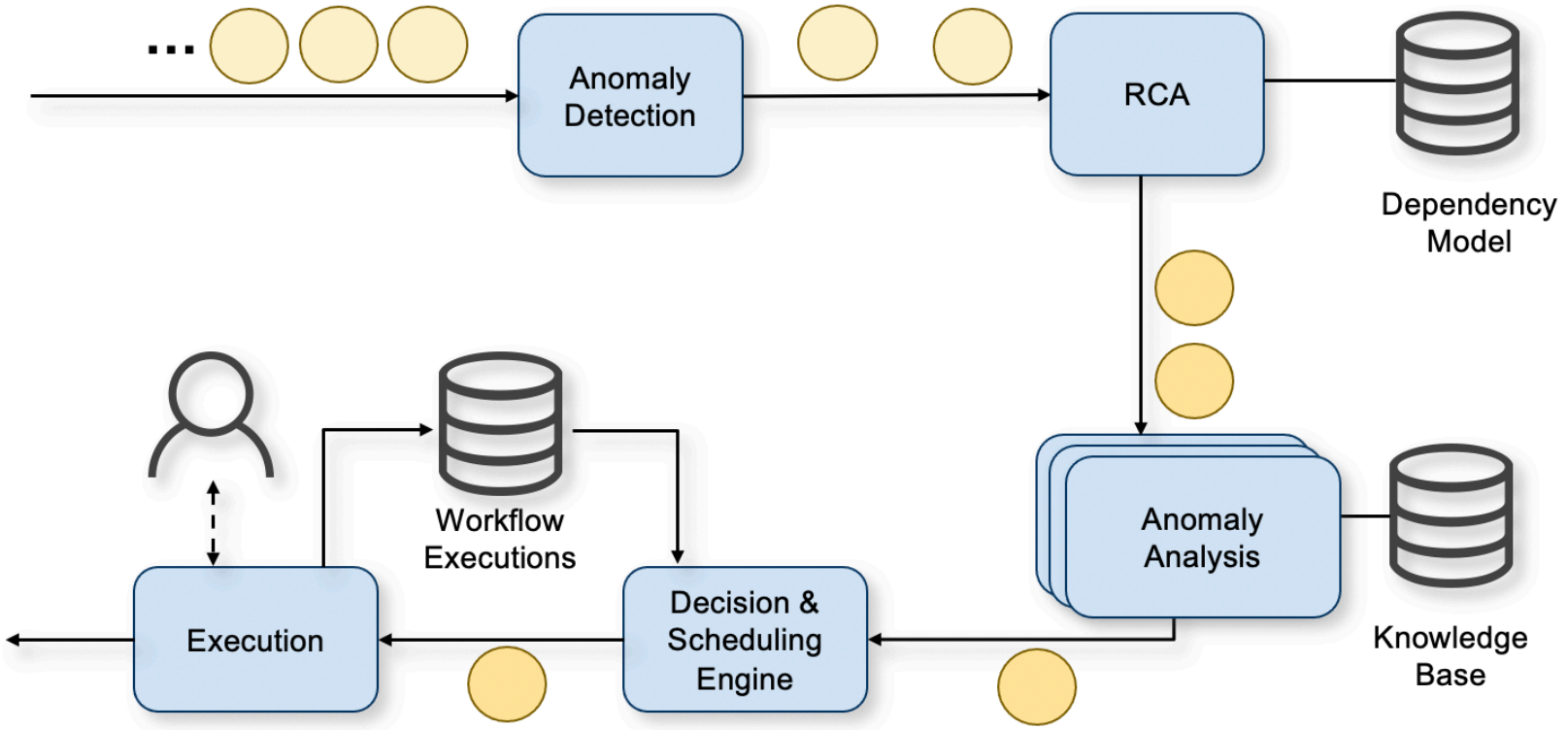
- On-Call rotations for operators to be available at night and during weekends
- Tedious and repetitive tasks during regular operation
- Extreme stress and high pressure in case of errors



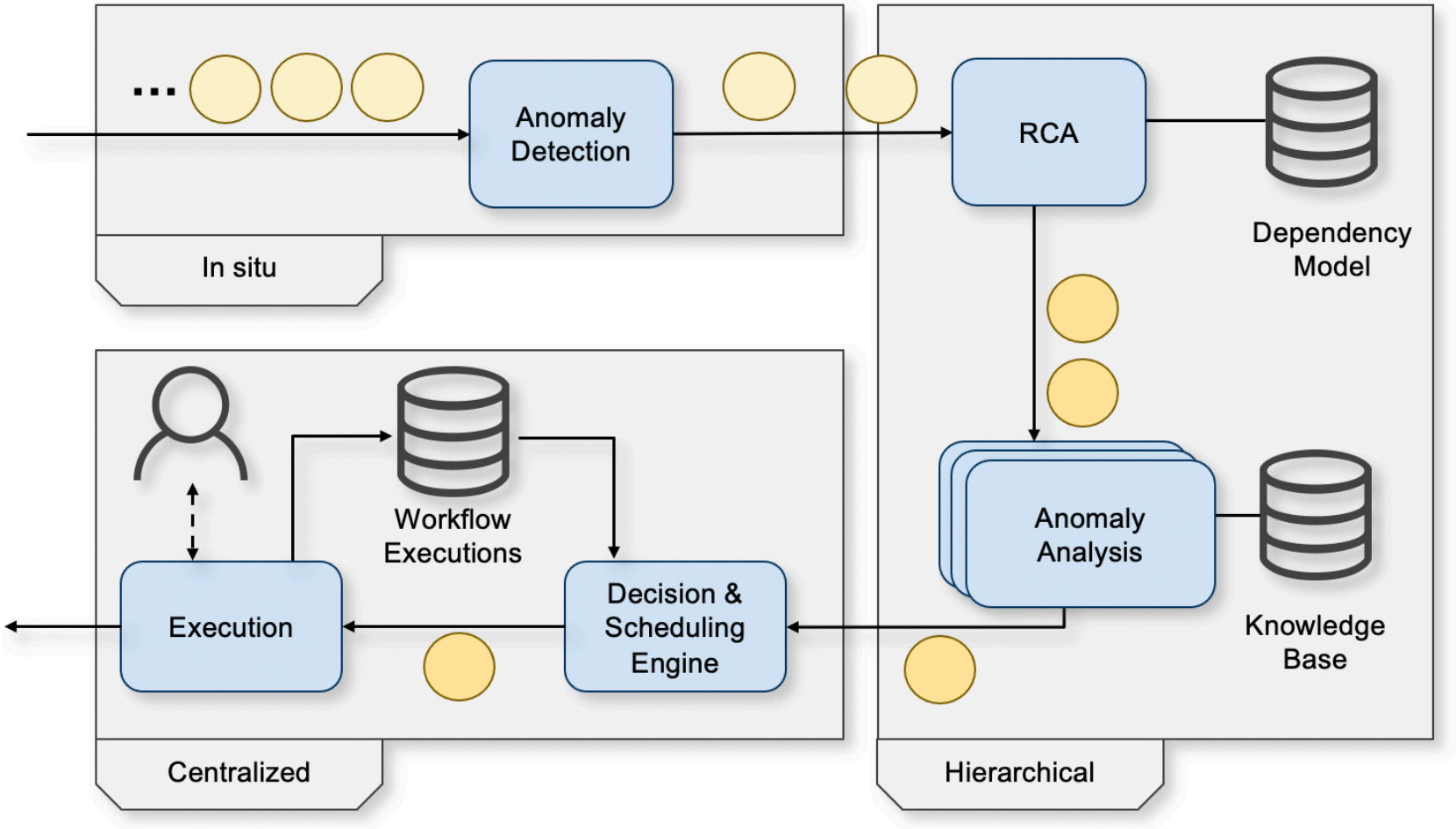
AIOps Market



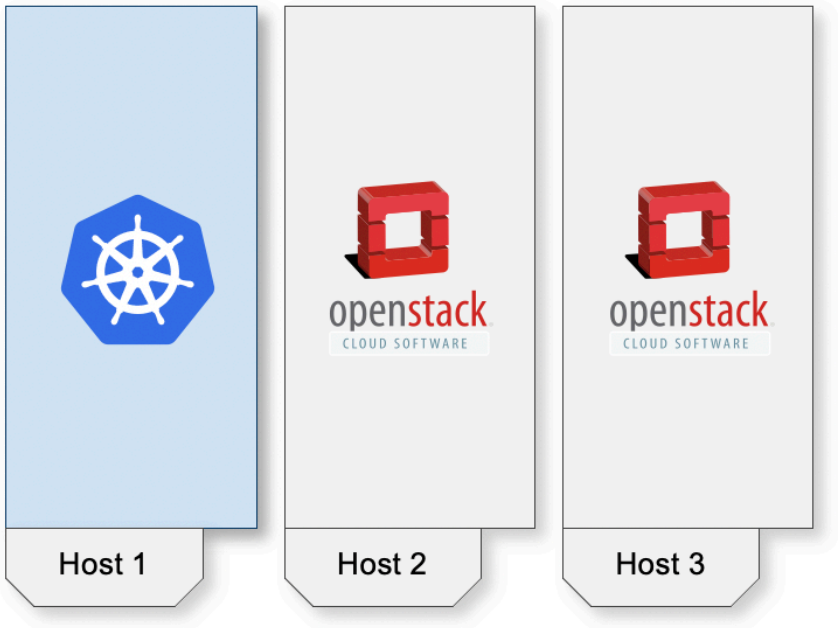
AIOps Components



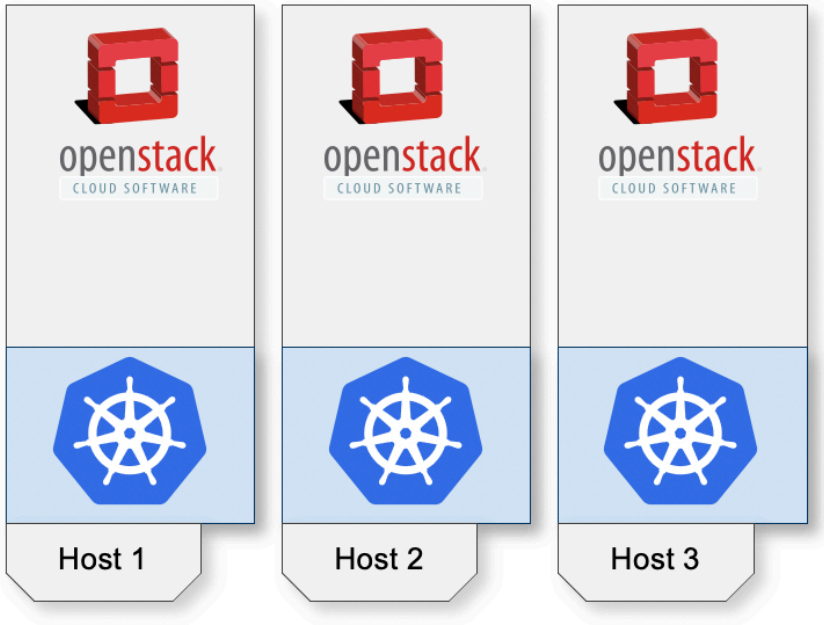
AIOps Components



In Situ Data Analysis

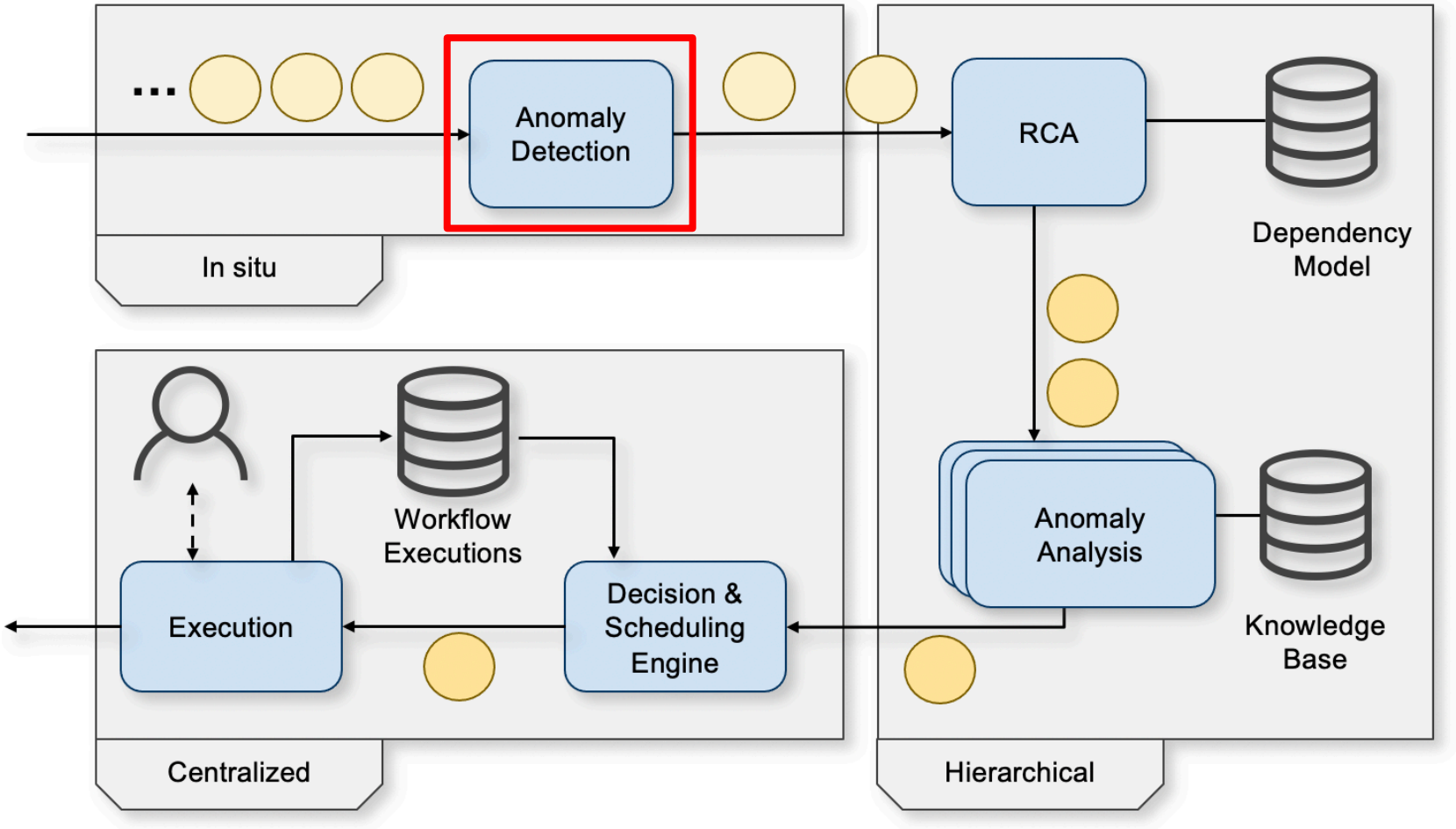


Dedicated data analysis hosts

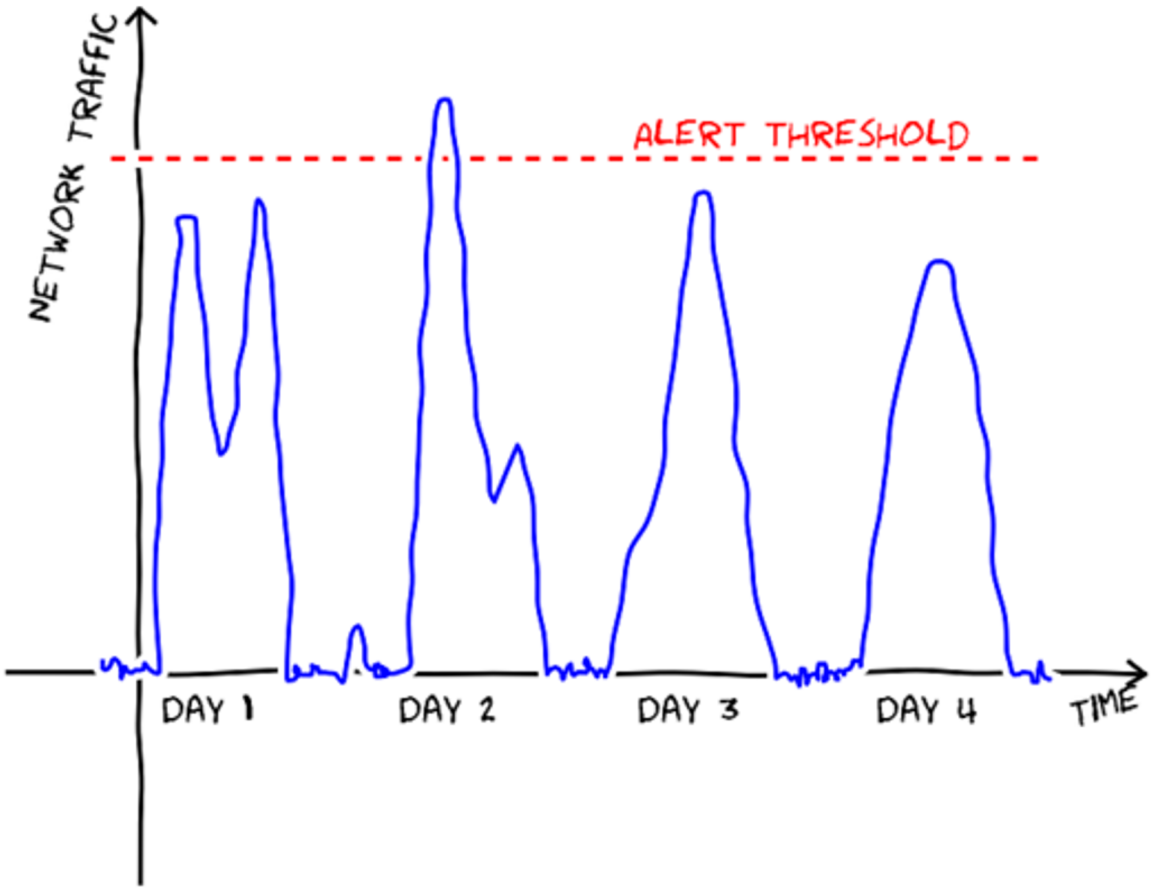


In situ data analysis

AIOps Components

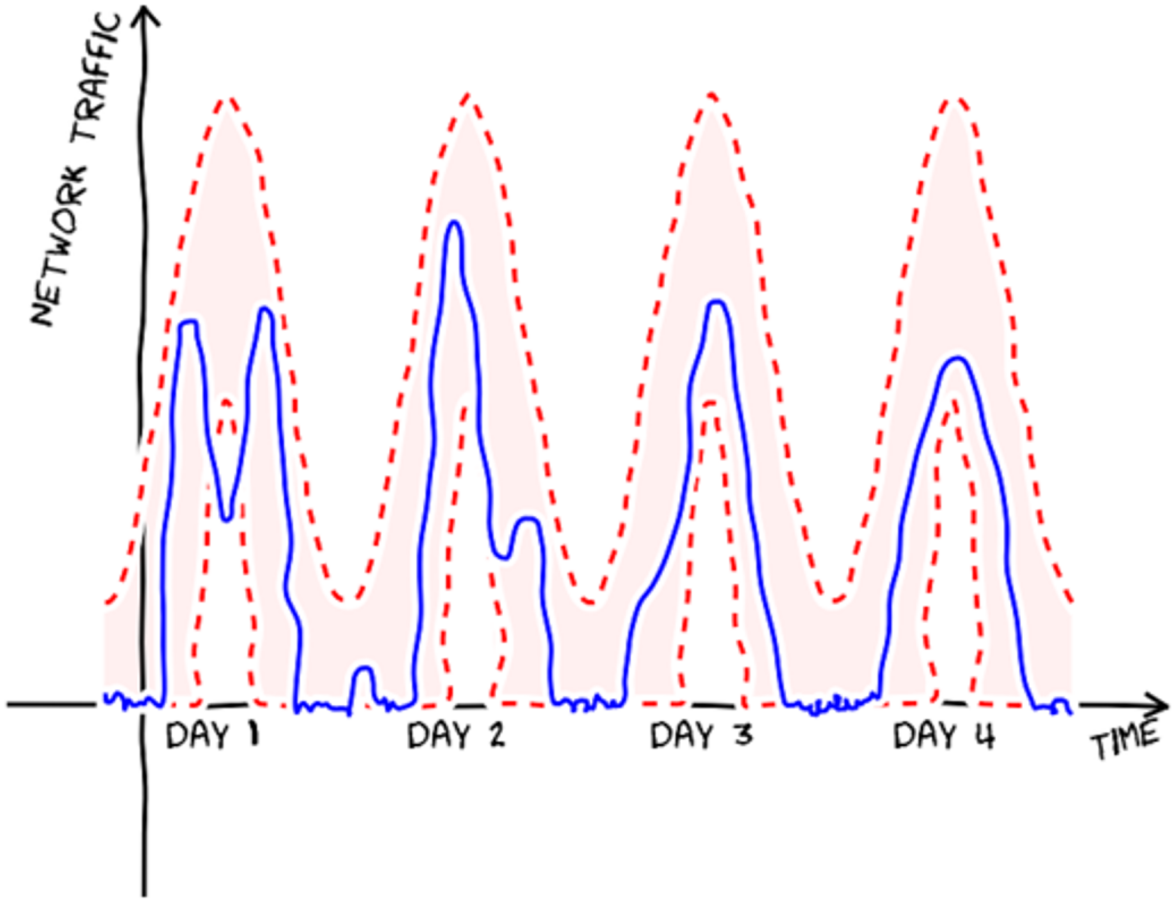


What is **unexpected** behaviour?



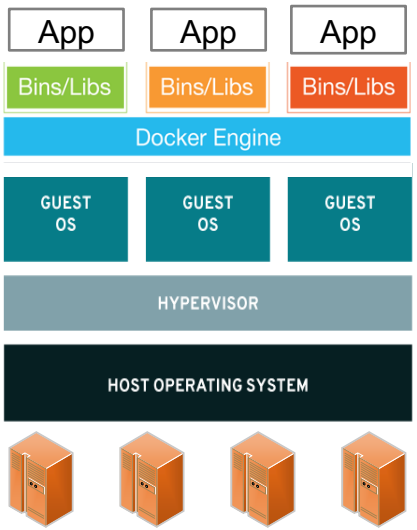
https://futurice.com/blog/tech-pick-of-the-week-log-anomaly-detection-tools

What is **unexpected** behaviour?



https://futurice.com/blog/tech-pick-of-the-week-log-anomaly-detection-tools

IFTM: Unsupervised Anomaly Detection

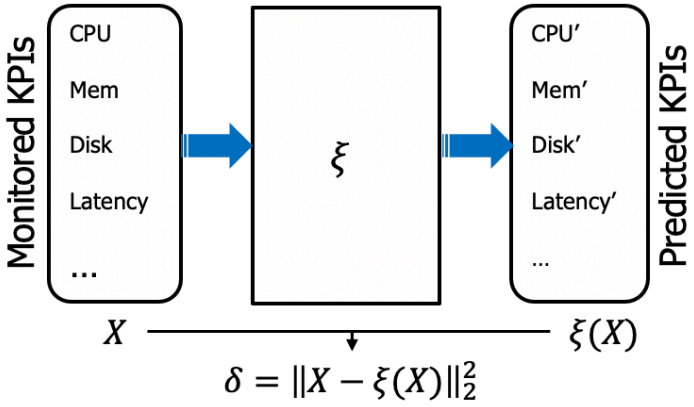


SaaS, IaaS, PaaS

↓

Monitoring KPIs on different layers

Identity **F**unction: Identity value prediction



Threshold **M**odel: Decision boundary τ for δ

Normal: $\delta \leq \tau$

Anomaly: $\delta > \tau$

ξ:

- Online ARIMA
- BIRCH, BICO
- Concept Adapting BIRCH
- Auto-Encoder (LSTM)

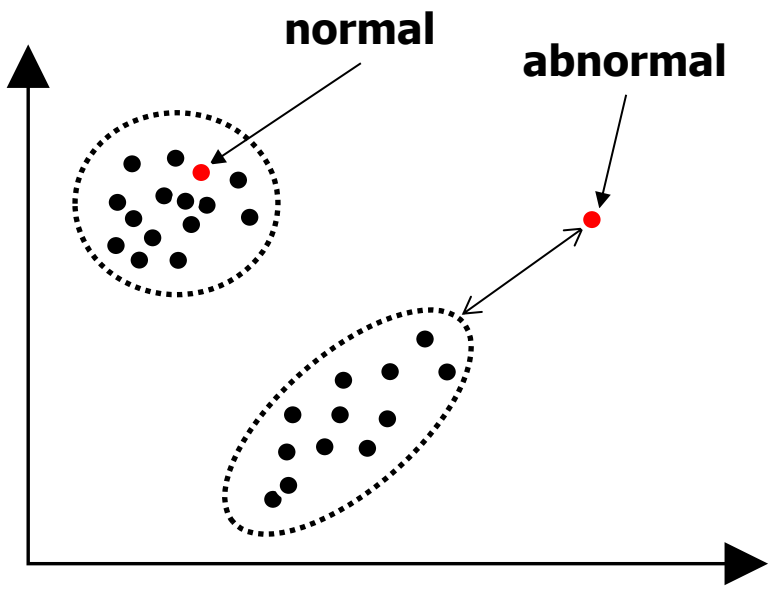
τ:

- Cumulative Aggregation
- Sliding Window Aggregation
- Exponential Moving Model
- Double Exponential Moving Model

Clustering based anomaly detection

BIRCH [1]: **B**alanced **I**terative **R**educing and **C**lustering using **H**ierarchies

- Unsupervised data mining algorithm
- Hierarchical clustering over very large data-sets
- Can be used for capturing the "normal" behavior

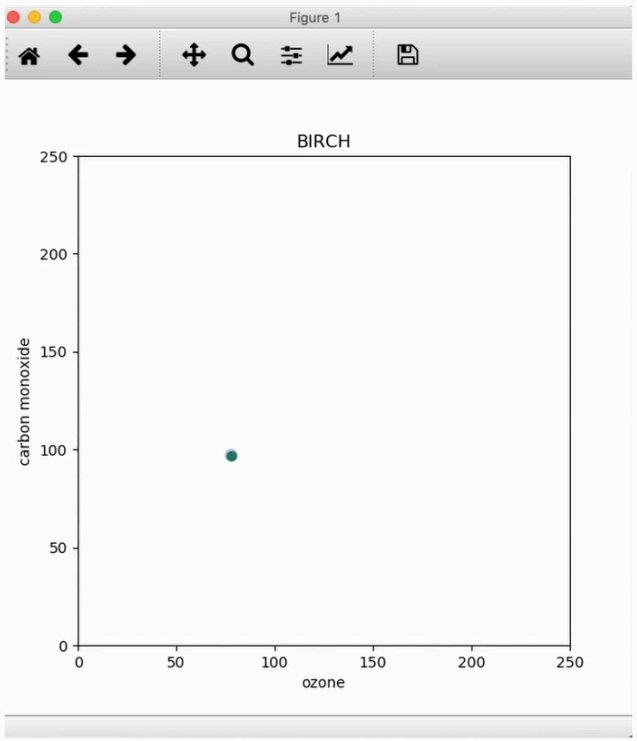


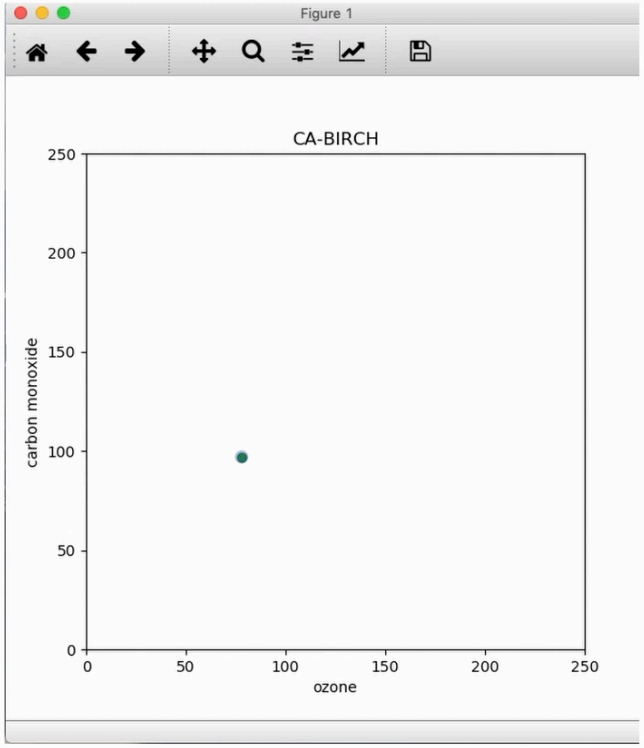
Problem: *Concept drifts* due to e.g. software aging

Solution: Letting BIRCH "forget" about the past

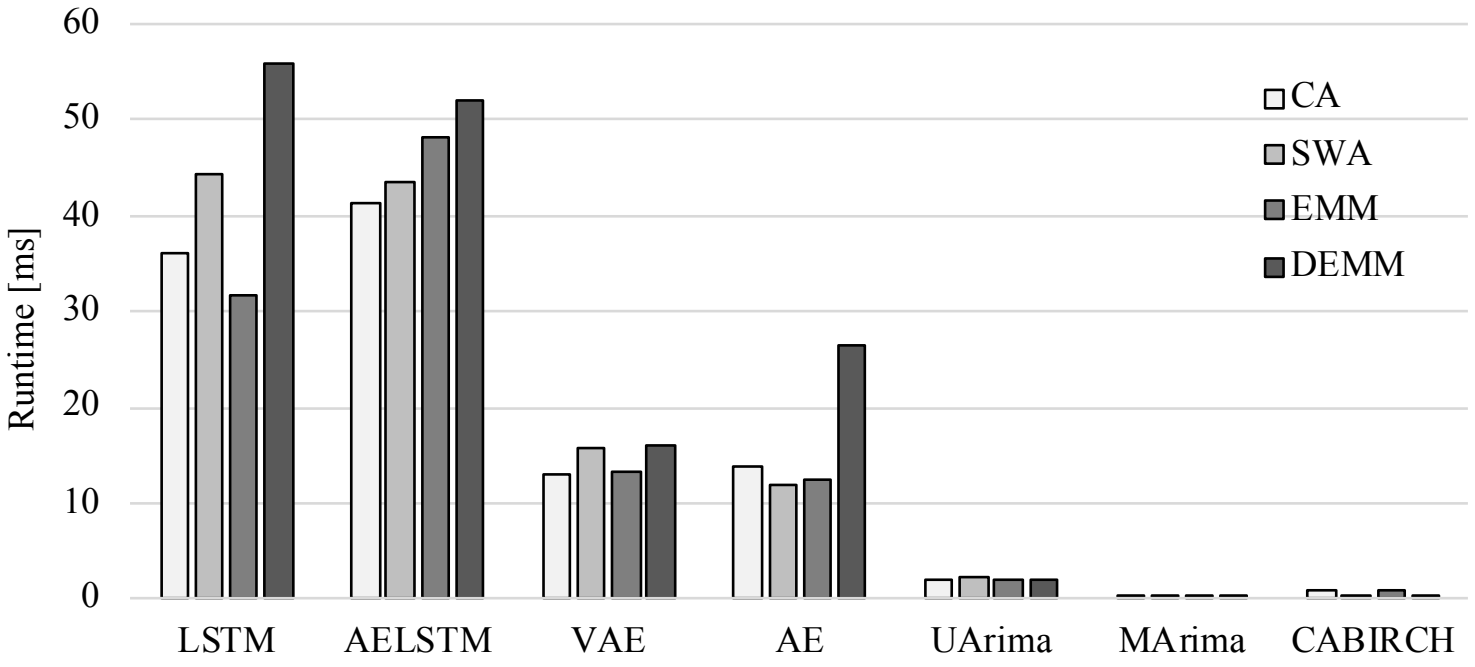
[1] Zhang et al. *BIRCH: an efficient data clustering method for very large databases*. SIGMOD (1996)

BIRCH and CA-BIRCH





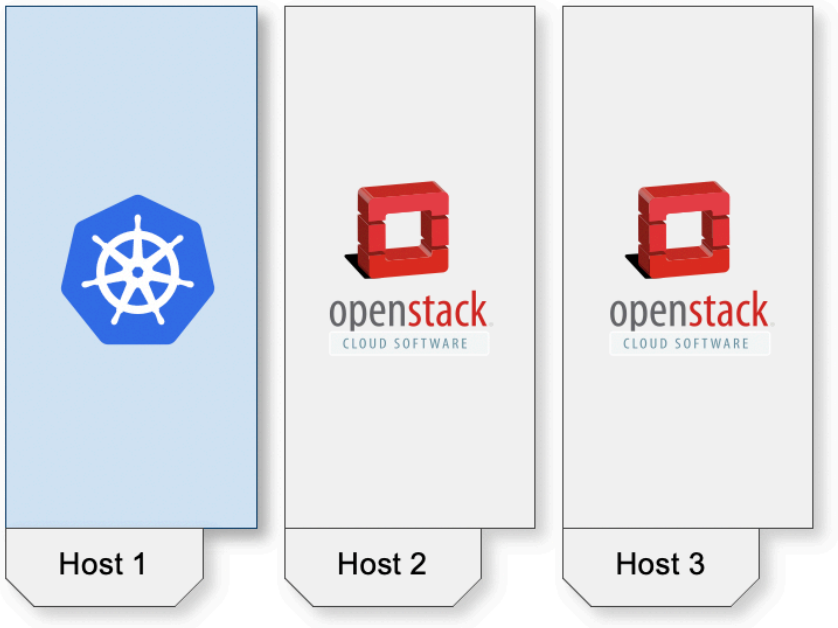
CABRICH for **in situ** analysis



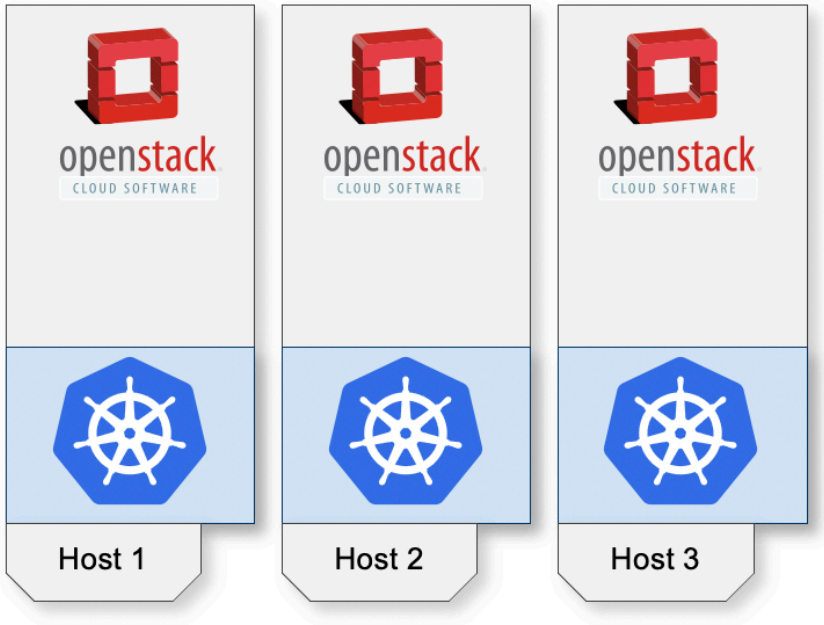
Deep Learning:
Best prediction performance

CABRICH:
Fast & low energy consumption

In Situ Data Analysis

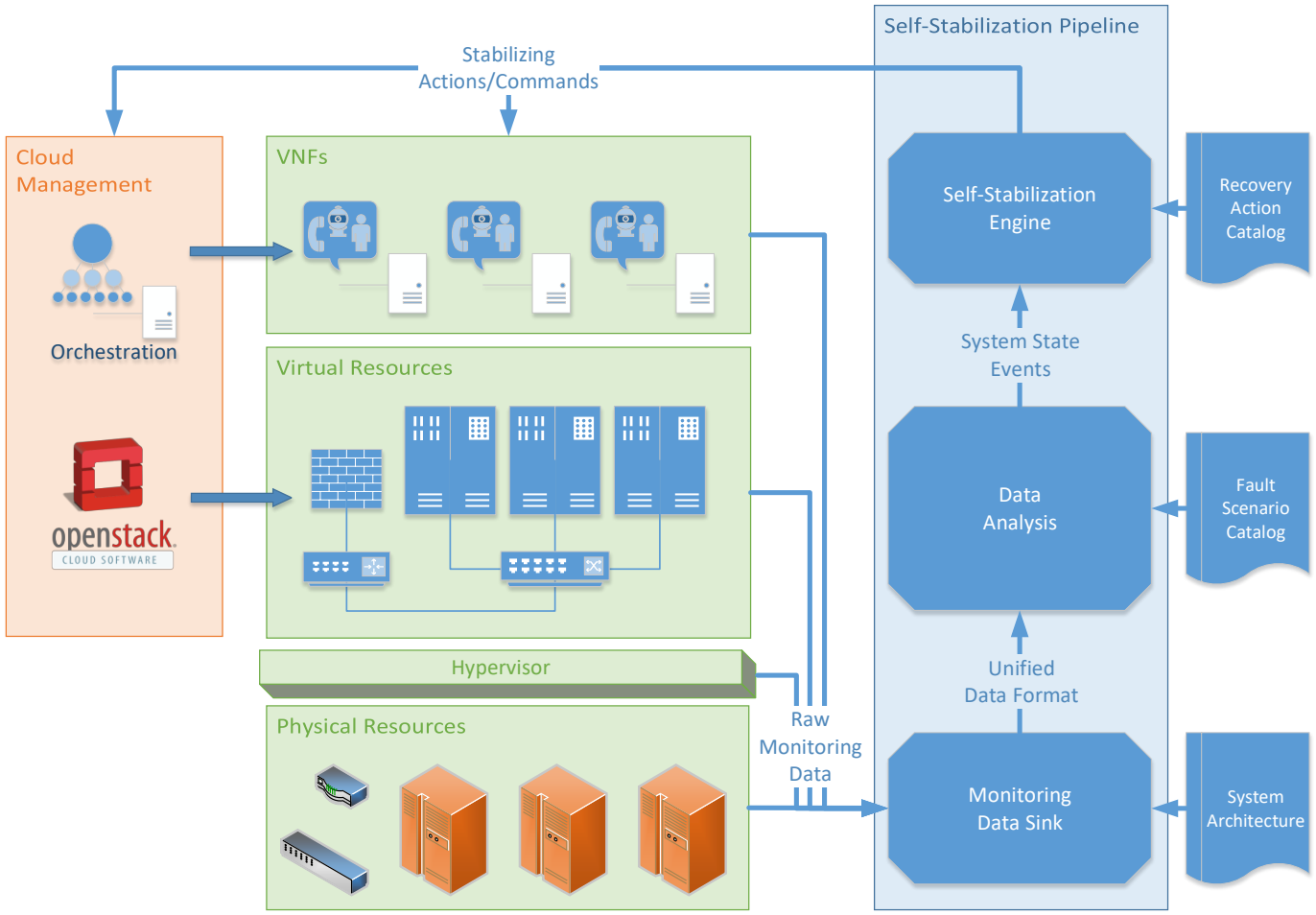


Dedicated data analysis hosts

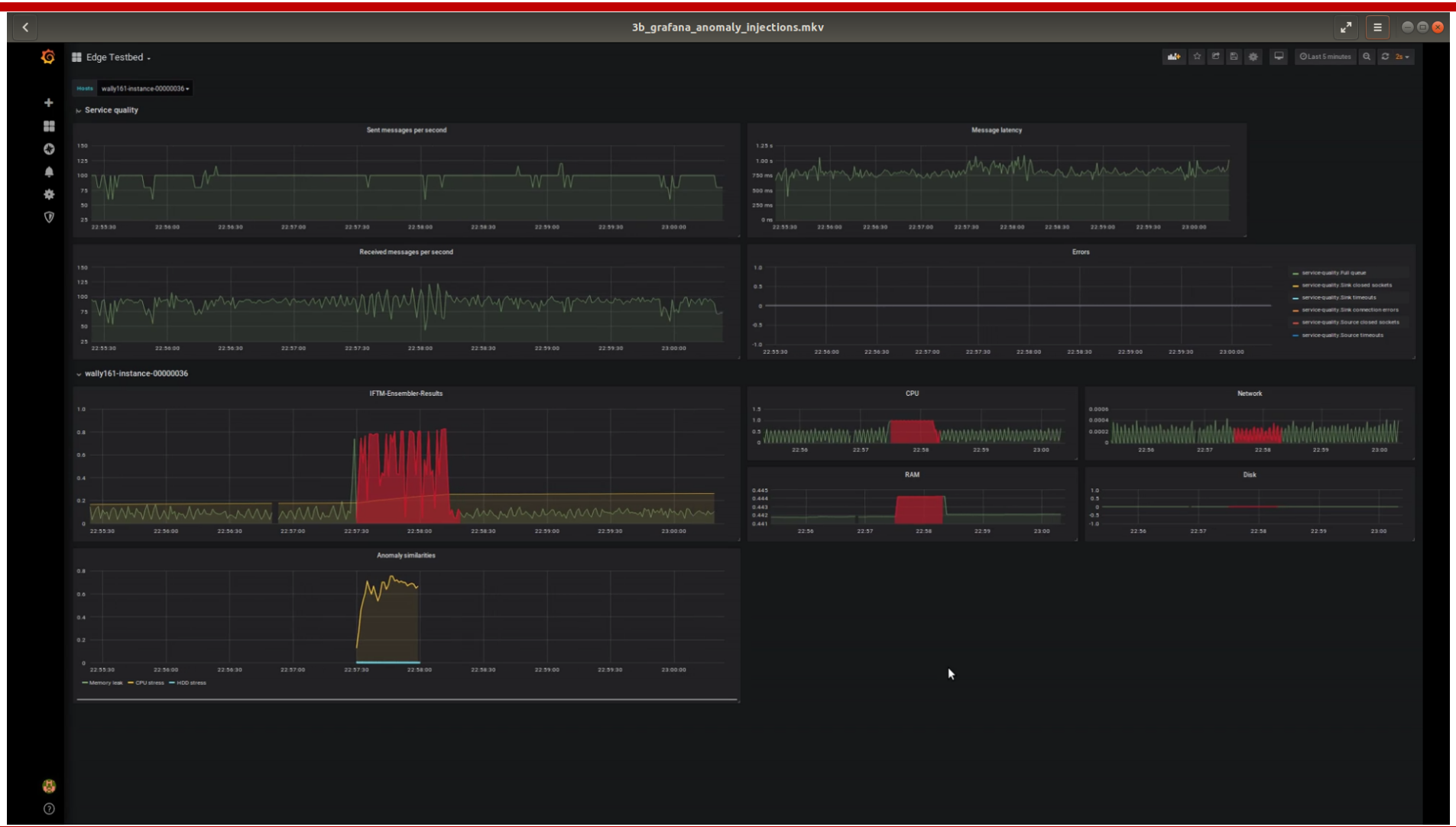


In situ data analysis

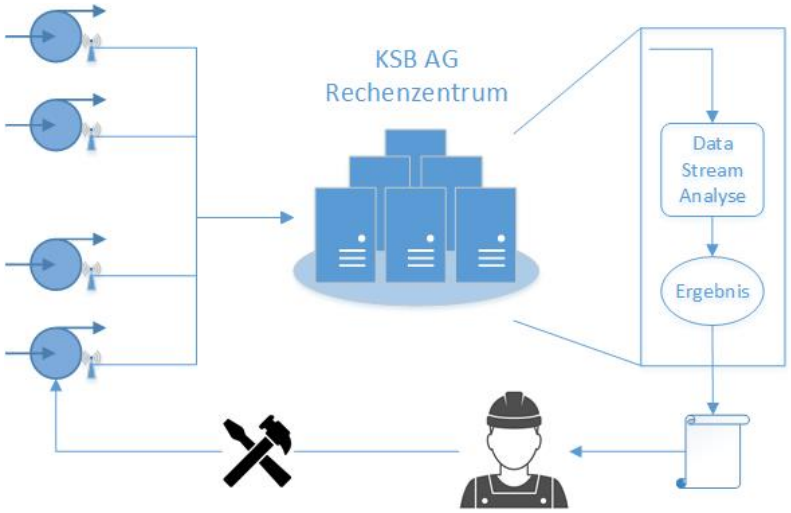
ZerOps: In Situ Data Analysis



ZerOps: In Situ Data Analysis

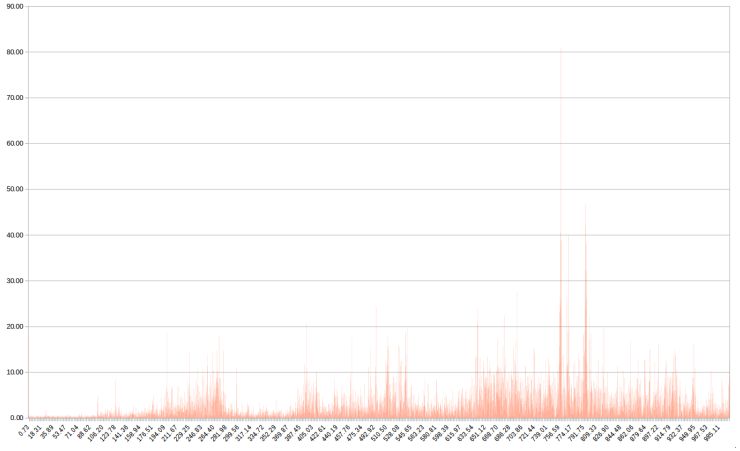


Industrial Water Pumps

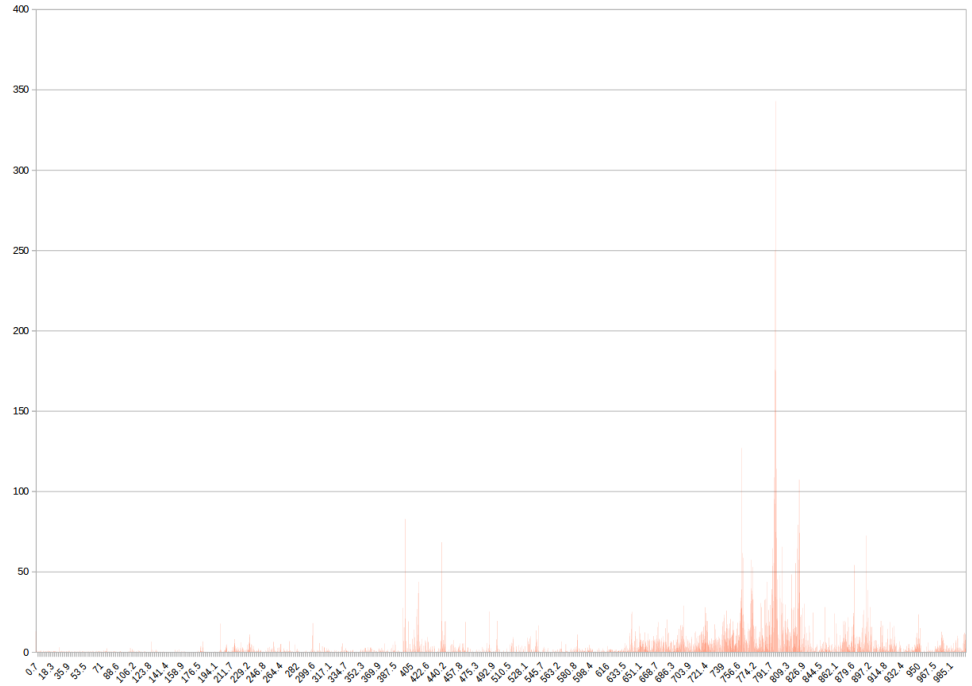


- Predictive Maintenance
- Increased reliability
- Reduction of costs

Abnormal Frequencies



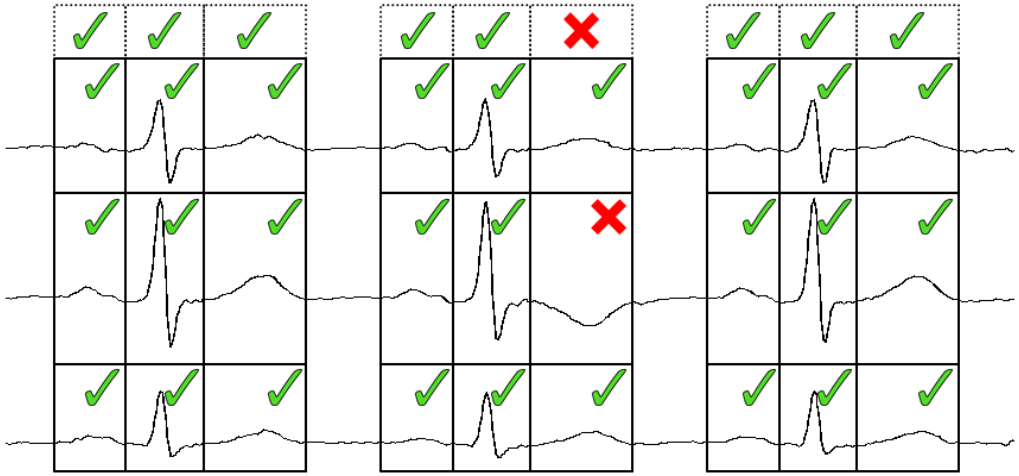
FFT



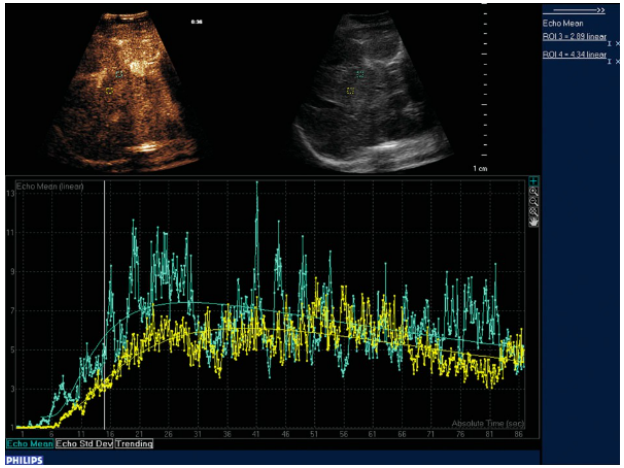
Distance to normal frequencies

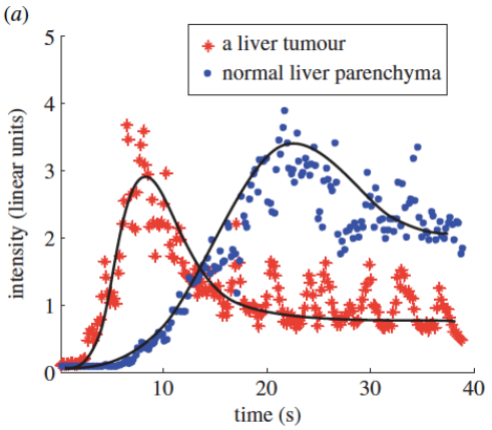
Medical analysis

ECG signal analysis



CEUS – Contrast Enhanced Ultrasound Videos





Summary

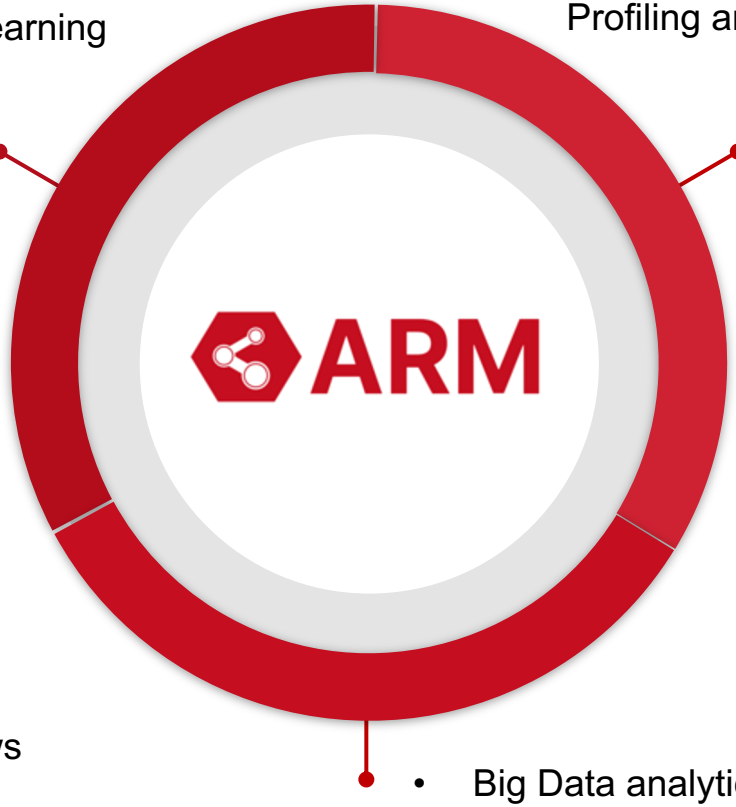
Main Goal: Operate distributed systems in a reliable, efficient and automated manner

We talked about:

- Self-optimizing and self-configuring systems
- Green computing through time series forecasting
- AIOps and self-healing systems
- Cutting-edge results in anomaly detection

Thank You <3

ARM: Fields of Research



Machine Learning

Profiling and performance modeling

Distributed Systems

- Data and task placement
- Batch and stream processing
- Internet of Things
- Scalability
- Fault Tolerance
- Dependability

Operating Systems

- Resource management
- Scheduling
- Data-parallel processing
- Virtual resources
- Real-time OS
- Networked embedded devices

Quality of service

Energy- and carbon-efficiency

Scientific workflows

Software Engineering

- Big Data analytics
- Emulations and Simulations
- System configuration
- Testing