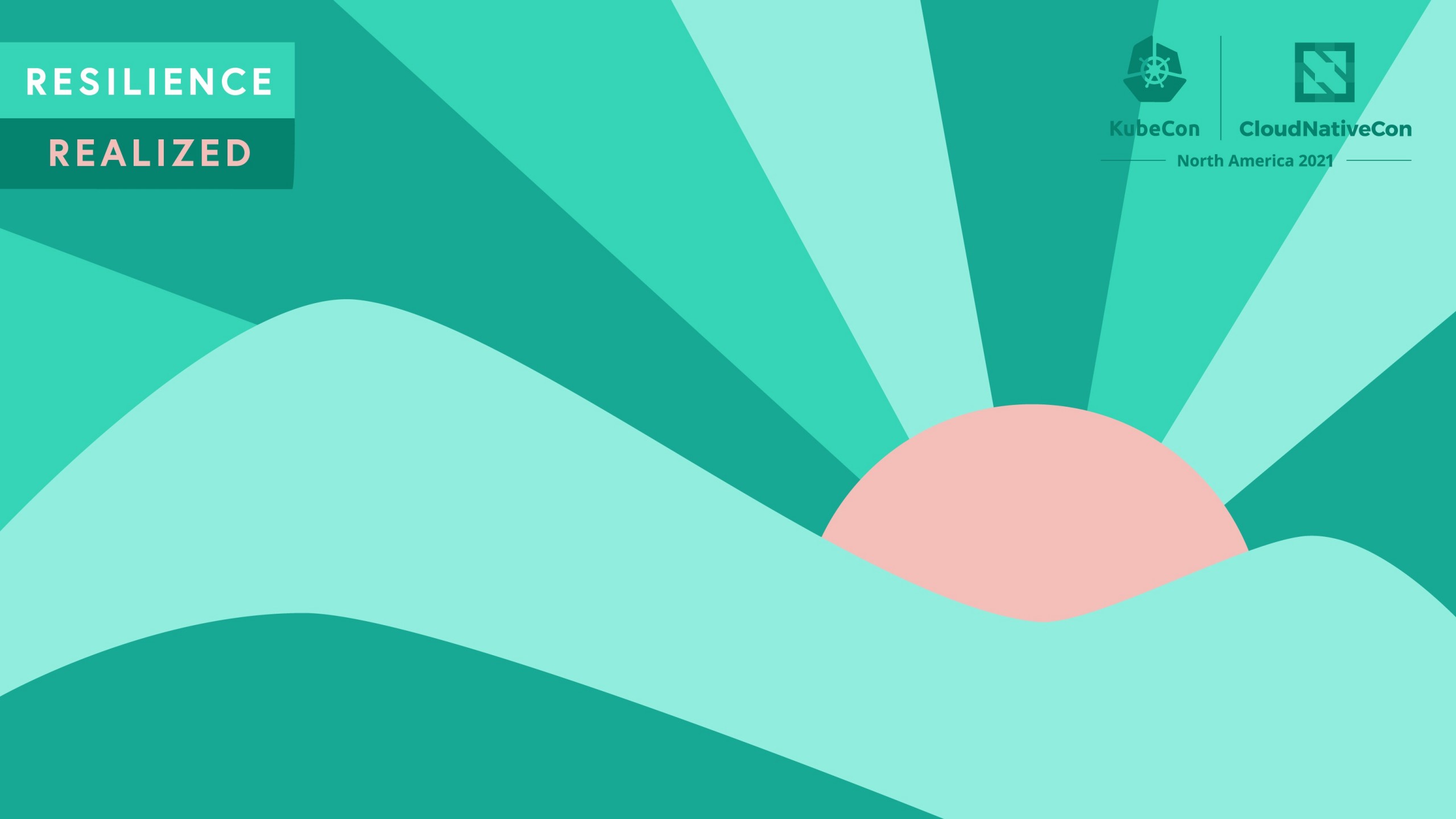
RESILIENCE REALIZED




KubeCon | CloudNativeCon

North America 2021

RESILIENCE

REALIZED

How to Improve Your K8s Experience with MLOps

Maksim Chudnovskii & Igor Gustomyasov, Sber

About Speakers

Igor Gustomyasov

Sber

Head of Integration Department

Russia

https://sberbank.ru/en/

Contact Me

on LinkedIn

Igor holds the position of Managing Director responsible for Synapse - service mesh-based target integration platform for Sberank and its growing ecosystem. Before this Igor was leading Center of Competence for integration solutions at Sberbank-Technology. During his career Igor was responsible for execution of large IT projects in financial sector, oil&gas industry, telco, government.

Maksim Chudnovskii

Sber

Chief Software Development Manager

Moscow, Russia

https://sberbank.ru/en/

Contact Me

on Twitter on LinkedIn

10 years of experience in software development and architecture, including 5 years in the banking industry as a developer, product manager, solution architect, and system architect. Key areas of the experience: High load integration solutions, Distributed platform solutions based on cloud-native technologies, CI/CD solutions for monolithic and microservice architectures, AI&ML solutions.

About Sber

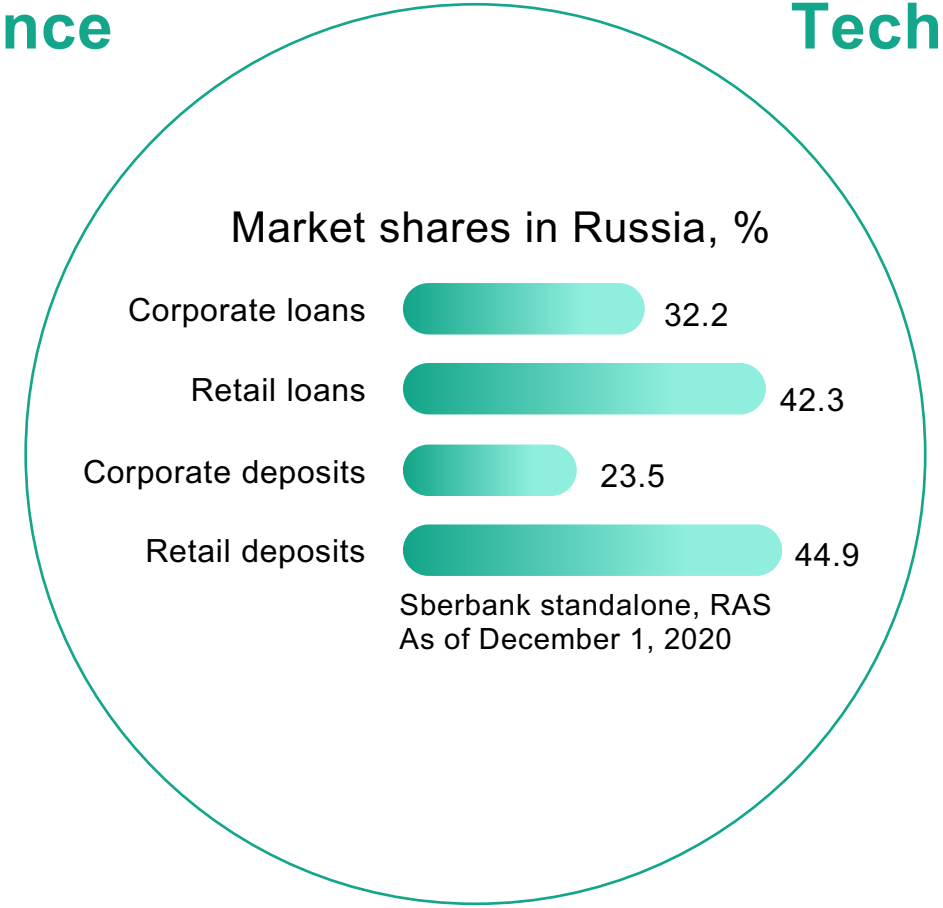
Best client experience

In financial services

98+ mn retail clients

2.7 mn corporates

The leader in digital services and sales



In non-financial services by 2023

10+ mn SberPrime subscribers

~500+ bn RUB e-commerce GMV

Technological leadership

New IT Platform

Reliability 99.99%

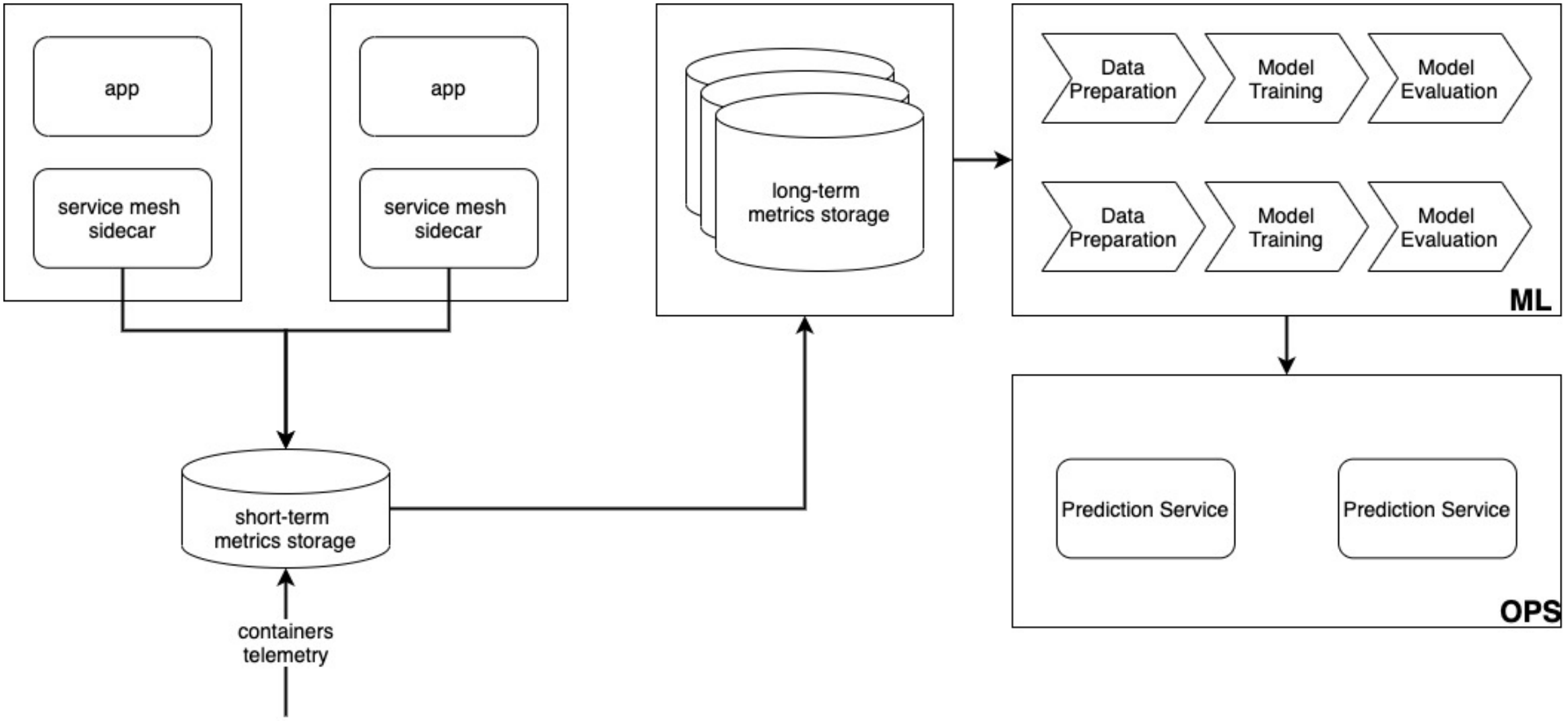
0 loses, **0** downtime

AI Platform launched and gives significant additional revenues and cost savings

Why AI?

- Many workloads have been migrated to K8s and Service Mesh and a huge amount of app telemetry has been collected
- Monitoring issues due to large & complex IT-systems landscape in a private cloud (alerts & metrics hell, unclear microservices topology and dependencies, etc.)
- Elasticity issues (inefficient HPA due to high apps startup time)
- Product issues (high overall latency due to network communication)

High Level Concept



Workloads Colocation

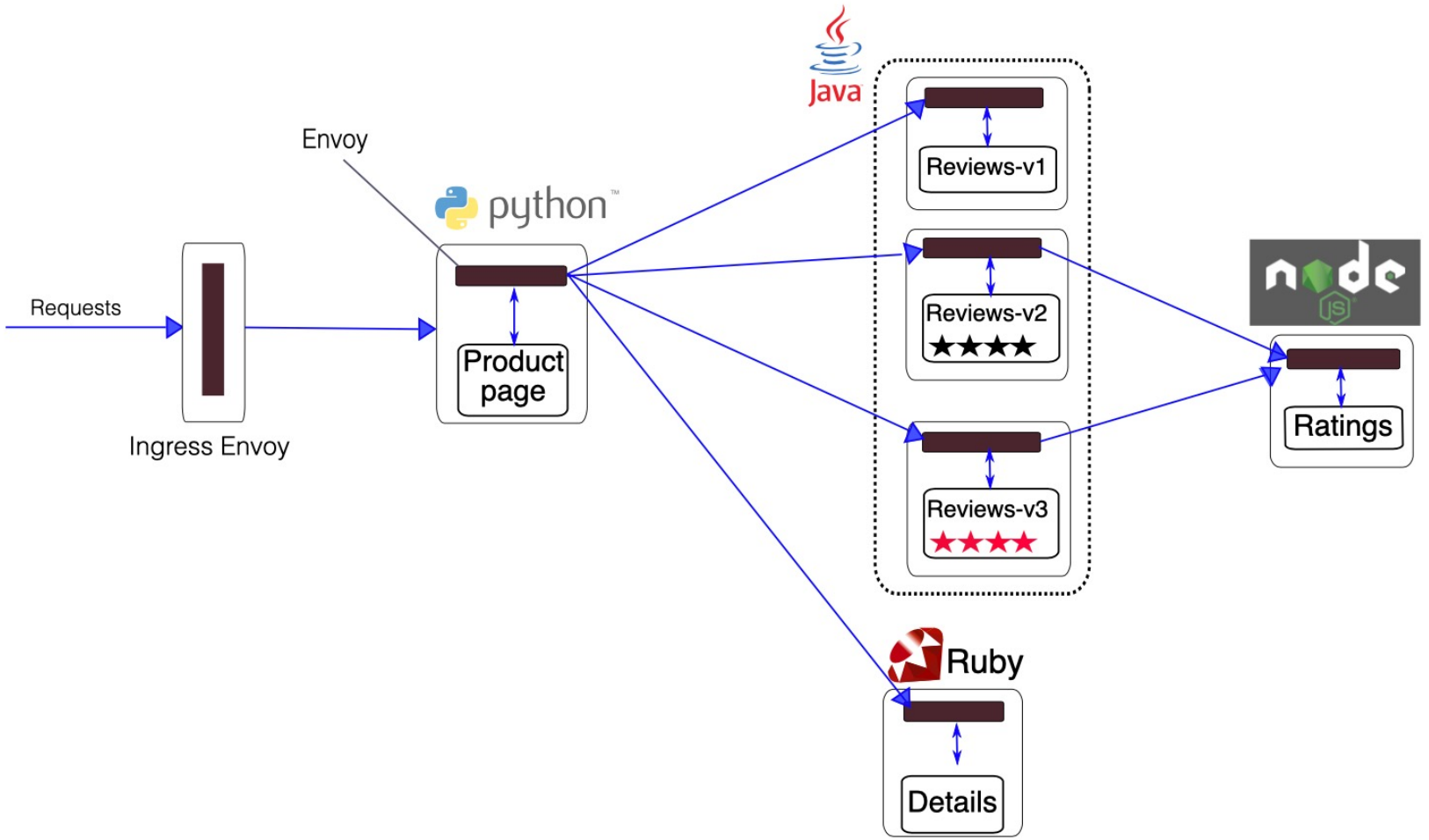
Assumption

- We can reduce network resource consumption as well as optimize overall latency by combining workloads in the ”schedule groups”

Metrics list

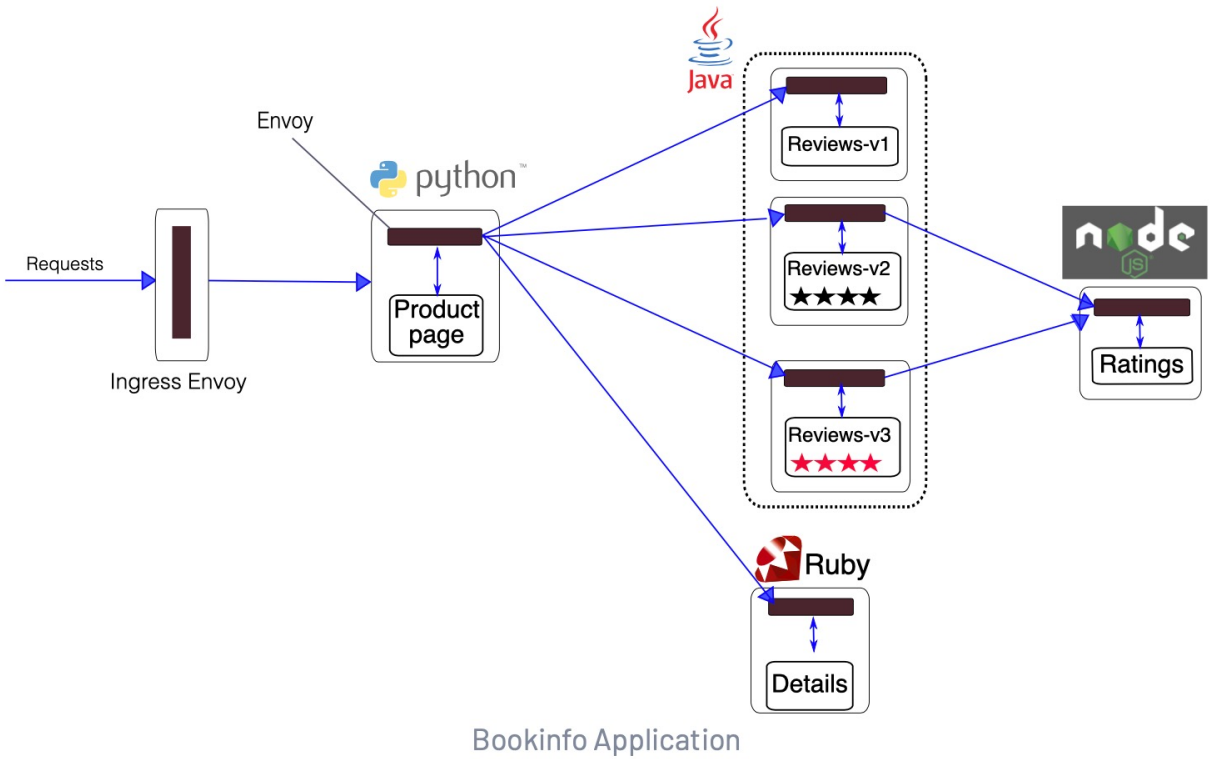
- istio_requests_total
- istio_request_duration_seconds_bucket
- istio_request_duration_seconds_count
- istio_request_duration_seconds_sum

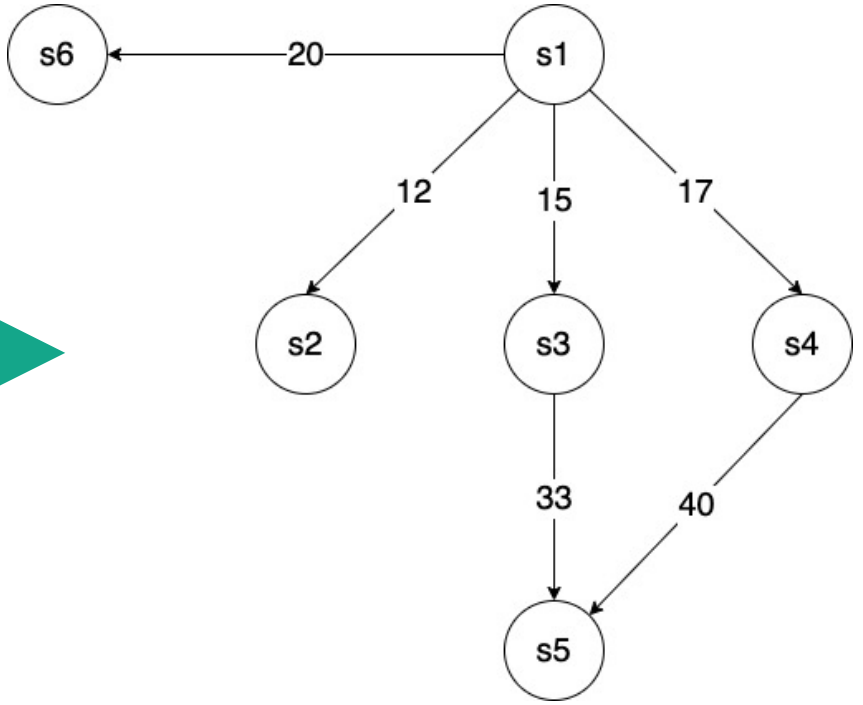
Workloads Colocation



Bookinfo Application

Workloads Colocation





Workloads Colocation

№	Source	Destination	Cost
1	s4	s5	40
2	s3	s5	33
3	s1	s6	20
4	s1	s4	17
5	s1	s3	15
6	s1	s2	12

Workloads Colocation

№	Source	Destination	Cost
1	s4	s5	40
2	s3	s5	33
3	s1	s6	20
4	s1	s4	17
5	s1	s3	15
6	s1	s2	12

Group	Services
1	s4 s5

Workloads Colocation

№	Source	Destination	Cost
1	s4	s5	40
2	s3	s5	33
3	s1	s6	20
4	s1	s4	17
5	s1	s3	15
6	s1	s2	12

Group	Services
1	s4 s5 s3

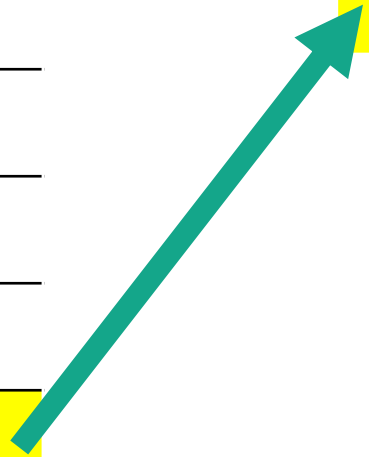
Workloads Colocation

№	Source	Destination	Cost
1	s4	s5	40
2	s3	s5	33
3	s1	s6	20
4	s1	s4	17
5	s1	s3	15
6	s1	s2	12

Group	Services
1	s4 s5 s3
2	s1 s6

Workloads Colocation

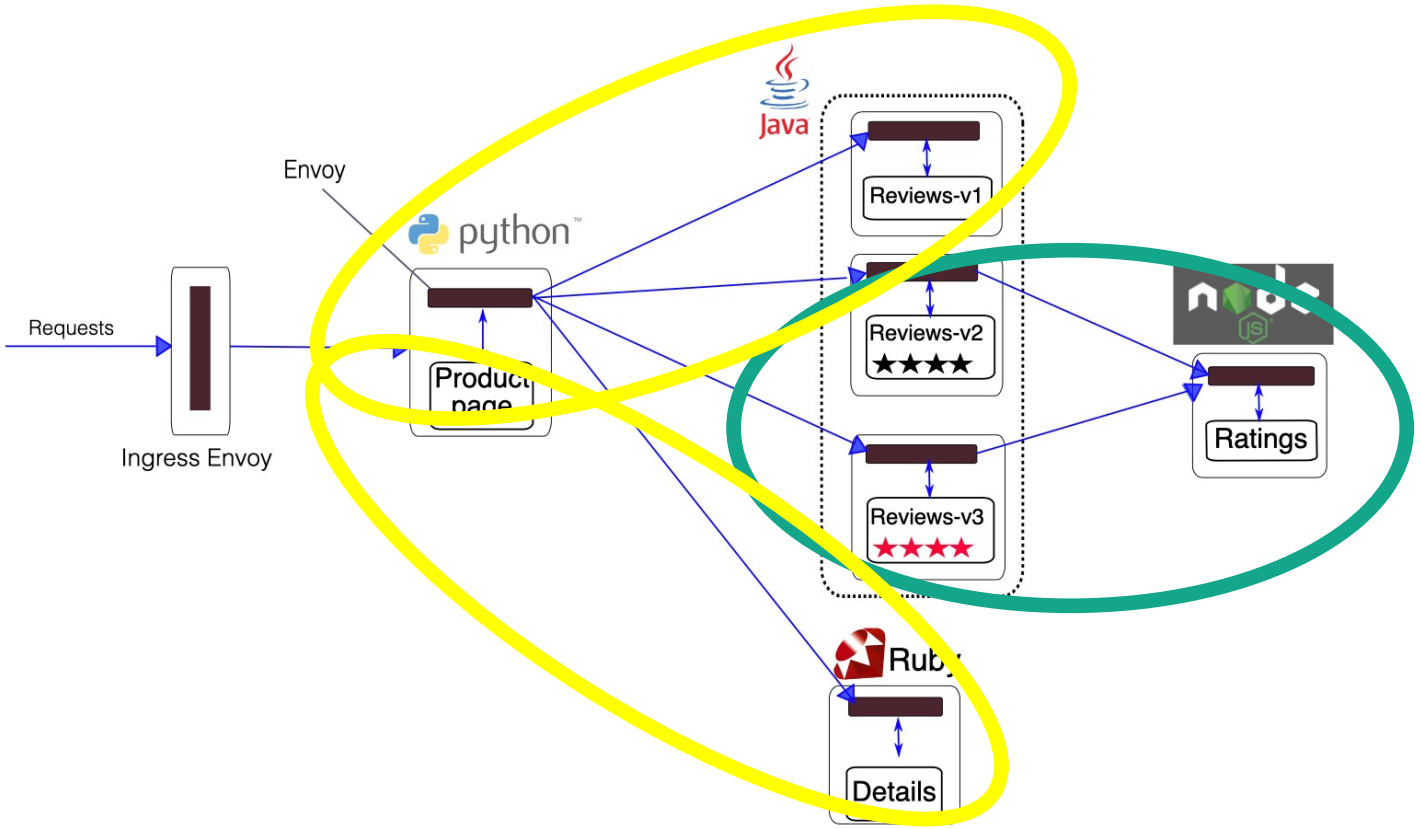
№	Source	Destination	Cost
1	s4	s5	40
2	s3	s5	33
3	s1	s6	20
4	s1	s4	17
5	s1	s3	15
6	s1	s2	12



Group	Services
1	s4 s5 s3
2	s1 s6 s2

Workloads Colocation

Group	Services
1	s4 s5 s3
2	s1 s6 s2



Bookinfo Application

Workloads Colocation

How to use it:

- Node Selectors
- Affinity Rules (pods & nodes)
- Topology Spread

What we do

- Latency & Network Consumption Optimization

Anomalies Detection

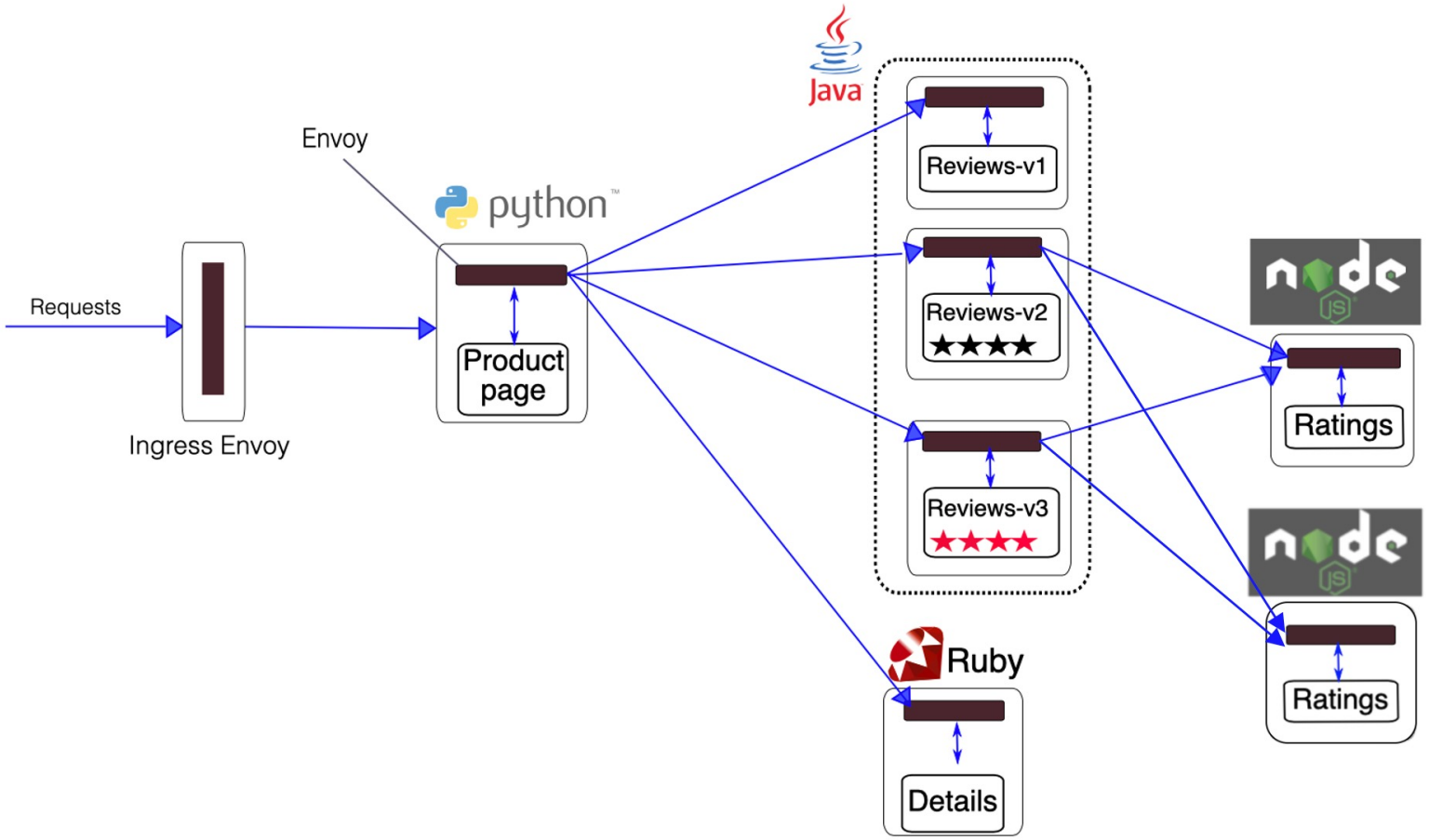
Assumption

- We can recognize abnormal values of a system and determine the root cause of these anomalies through the correlation of network and system metrics

Metrics list

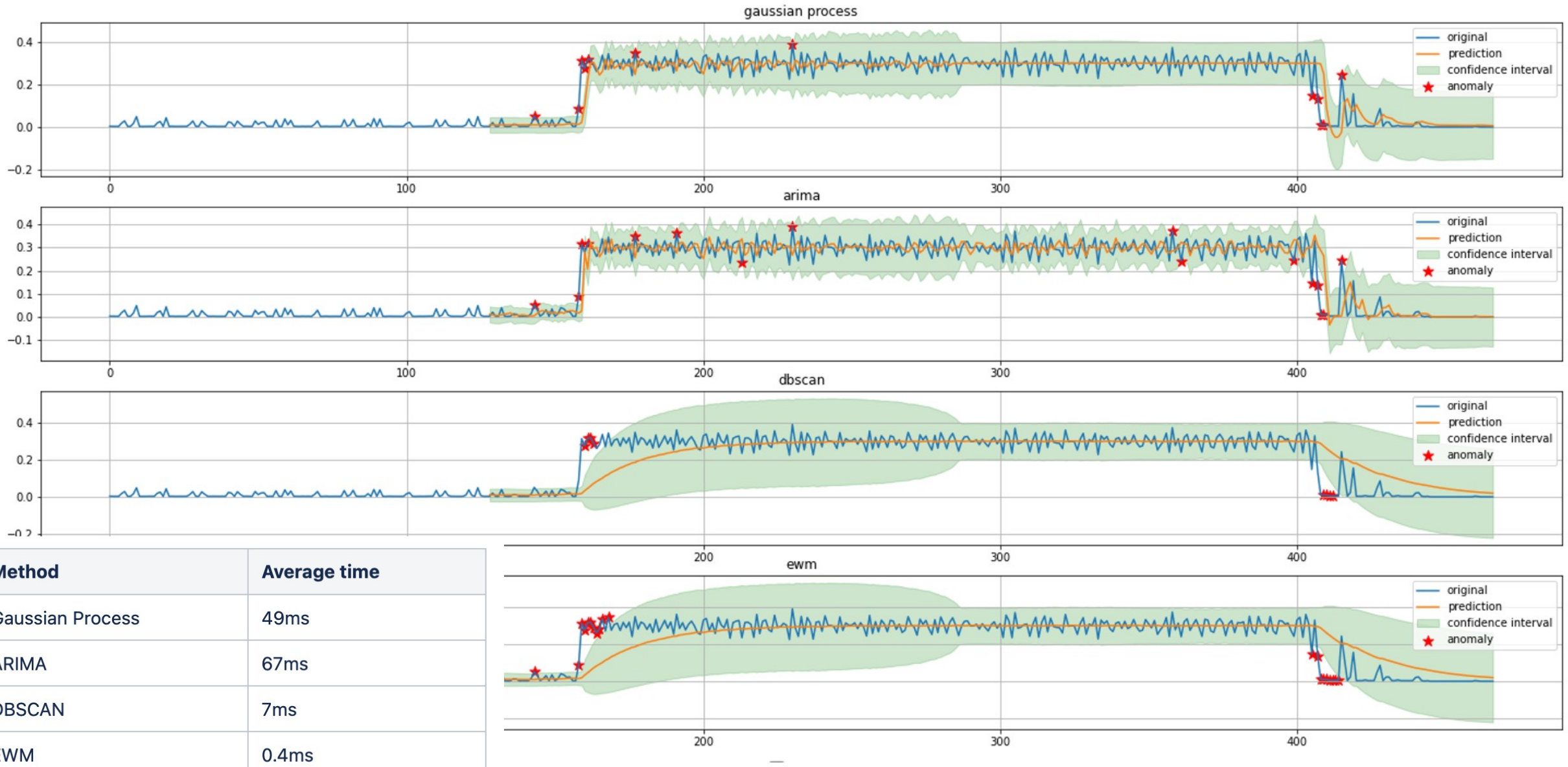
- container_sockets
- container_fs_inodes_total
- container_memory_cache
- container_memory_failcnt
- container_network_receive_errors_total
- container_network_transmit_errors_total
- container_network_transmit_packets_total
- container_fs_io_current
- container_fs_io_time_seconds_total
- container_fs_io_time_weighted_seconds_total
- container_fs_read_seconds_total
- container_fs_reads_merged_total
- container_spec_memory_swap_limit_bytes
- istio_tcp_sent_bytes_total
- container_cpu_usage_seconds_total
- container_memory_usage_bytes
- container_network_receive_bytes_total
- container_network_transmit_bytes_total
- container_memory_max_usage_bytes
- container_memory_failures_total
- container_memory_working_set_bytes
- container_memory_rss
- container_fs_usage_bytes
- container_cpu_system_seconds_total
- container_cpu_user_seconds_total
- container_fs_reads_total
- istio_request_duration_seconds_bucket

Anomalies Detection



Bookinfo Application

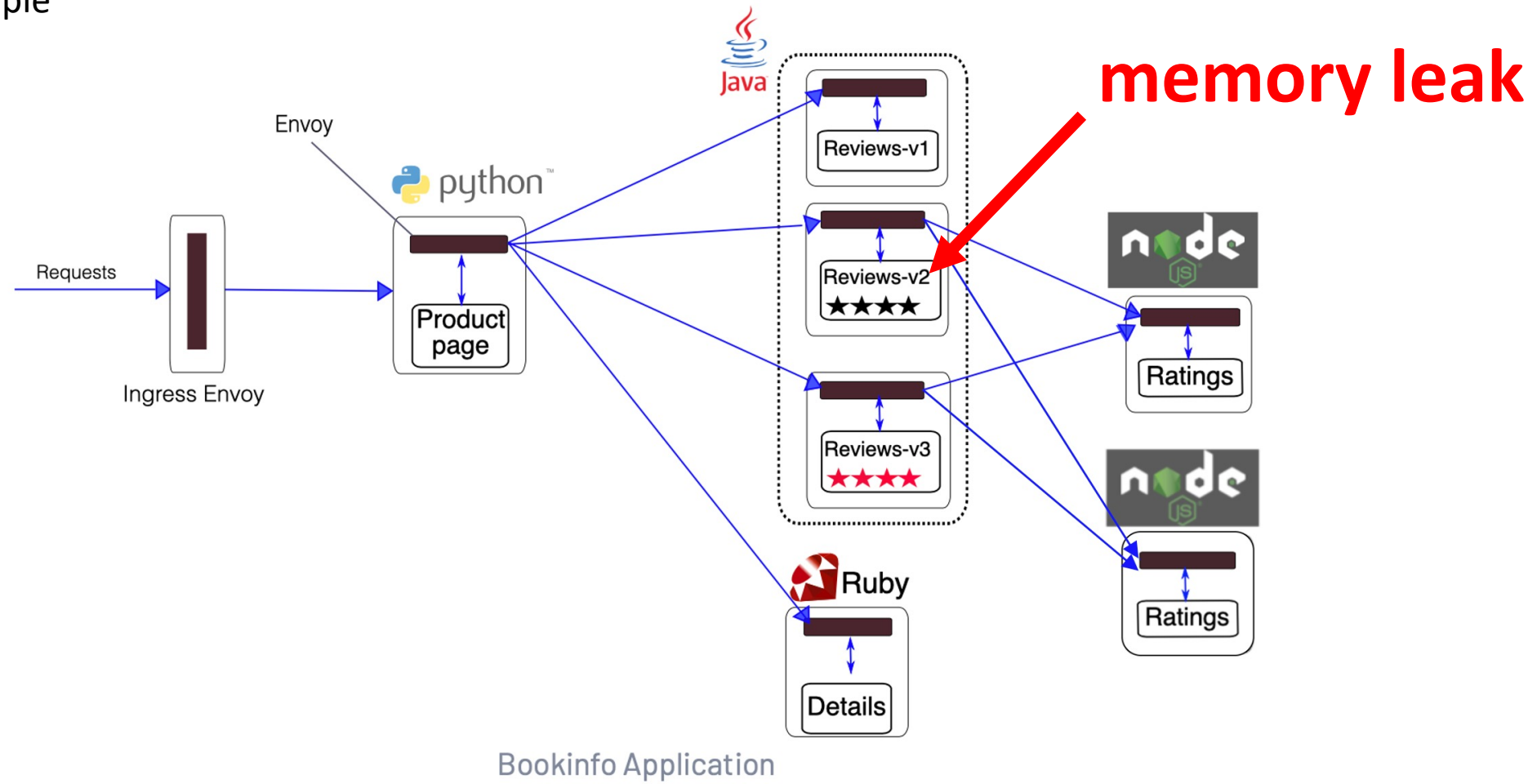
Anomalies Detection



Method	Average time
Gaussian Process	49ms
ARIMA	67ms
DBSCAN	7ms
EWM	0.4ms

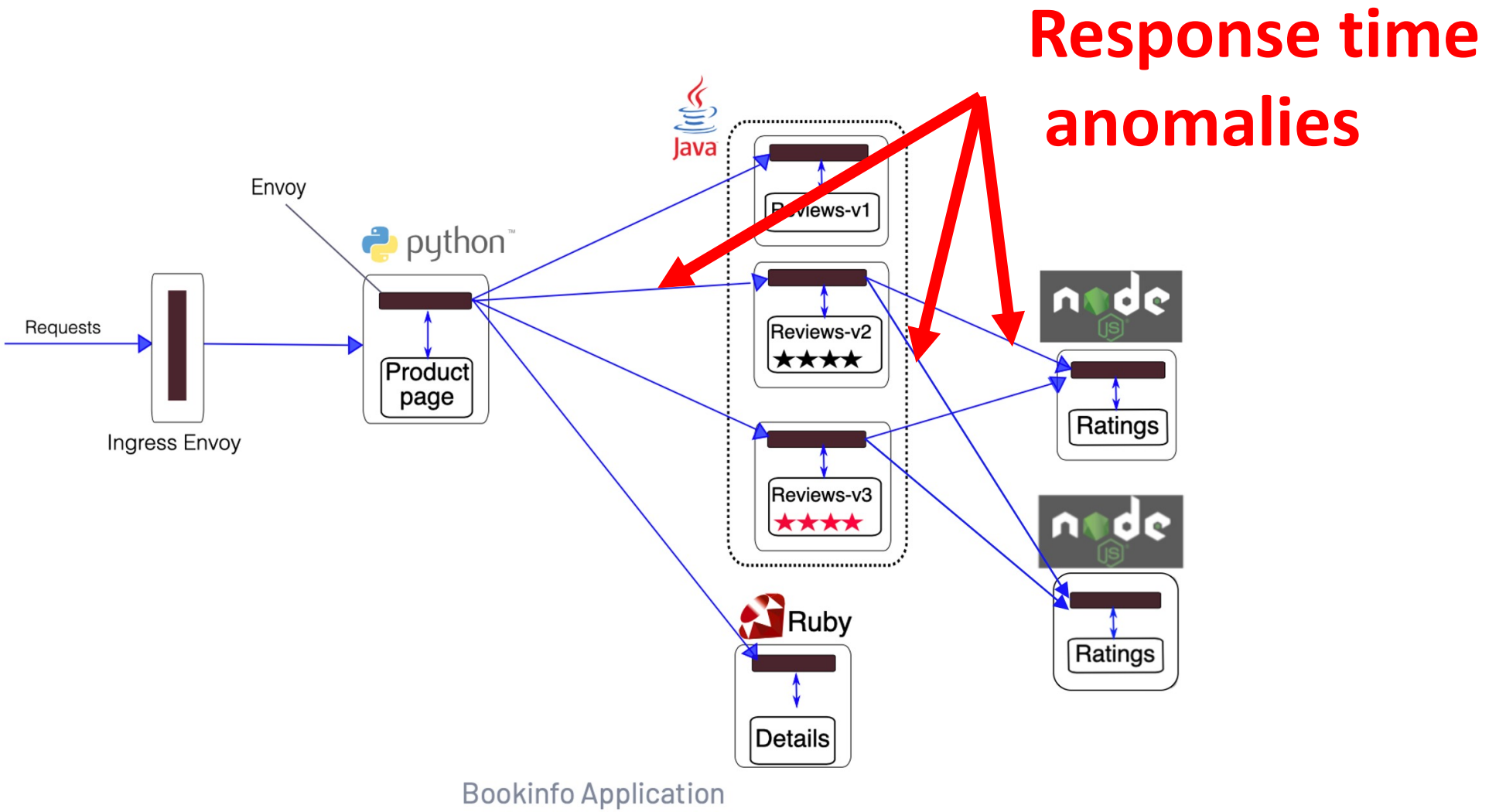
Anomalies Detection

MicroRCA[1] Example



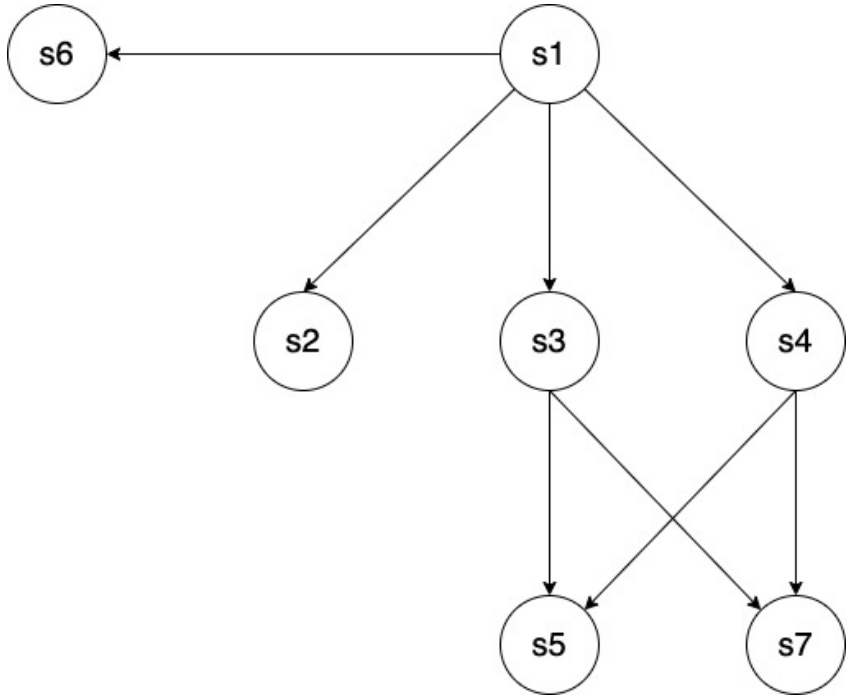
1 Li Wu, Johan Tordsson, Erik Elmroth, Odej Kao. MicroRCA: Root Cause Localization of Performance Issues in Microservices. IEEE/IFIP Network Operations and Management Symposium (NOMS), Apr 2020, Budapest, Hungary. hal-02441640

Anomalies Detection

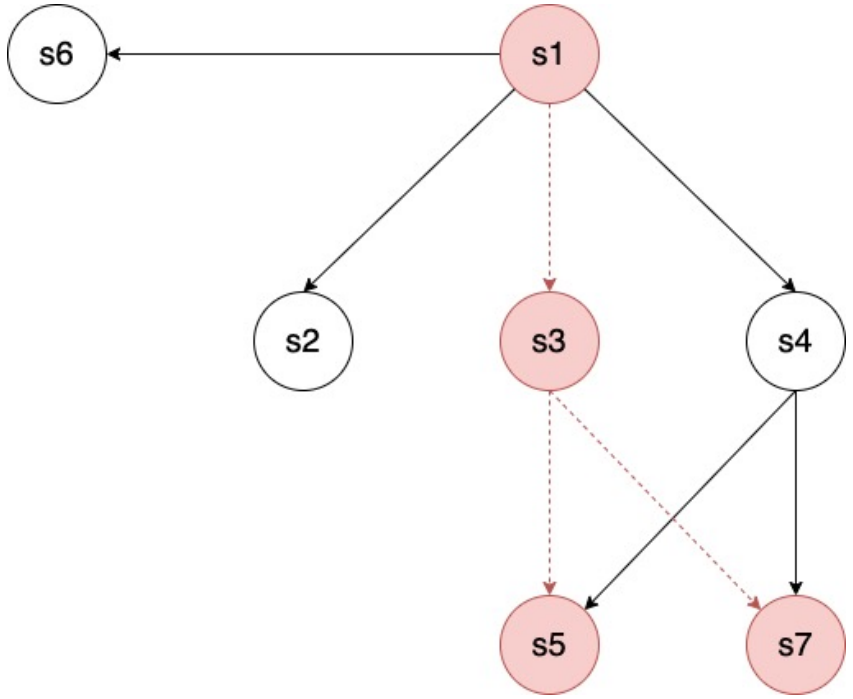


Anomalies Detection

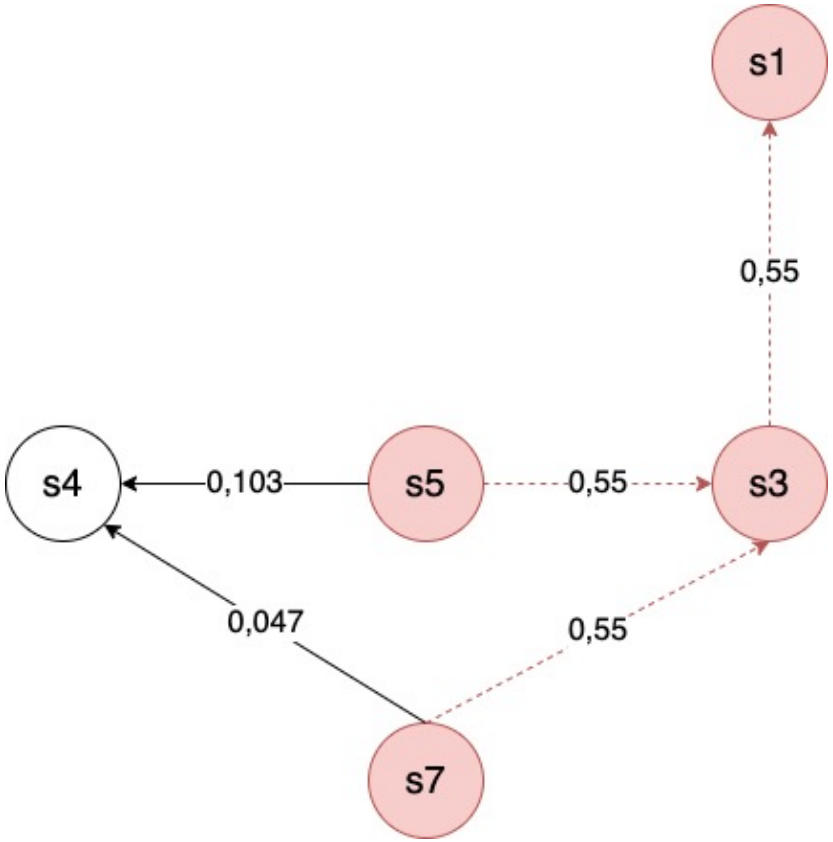
Services



Anomalies



Anomalies Detection



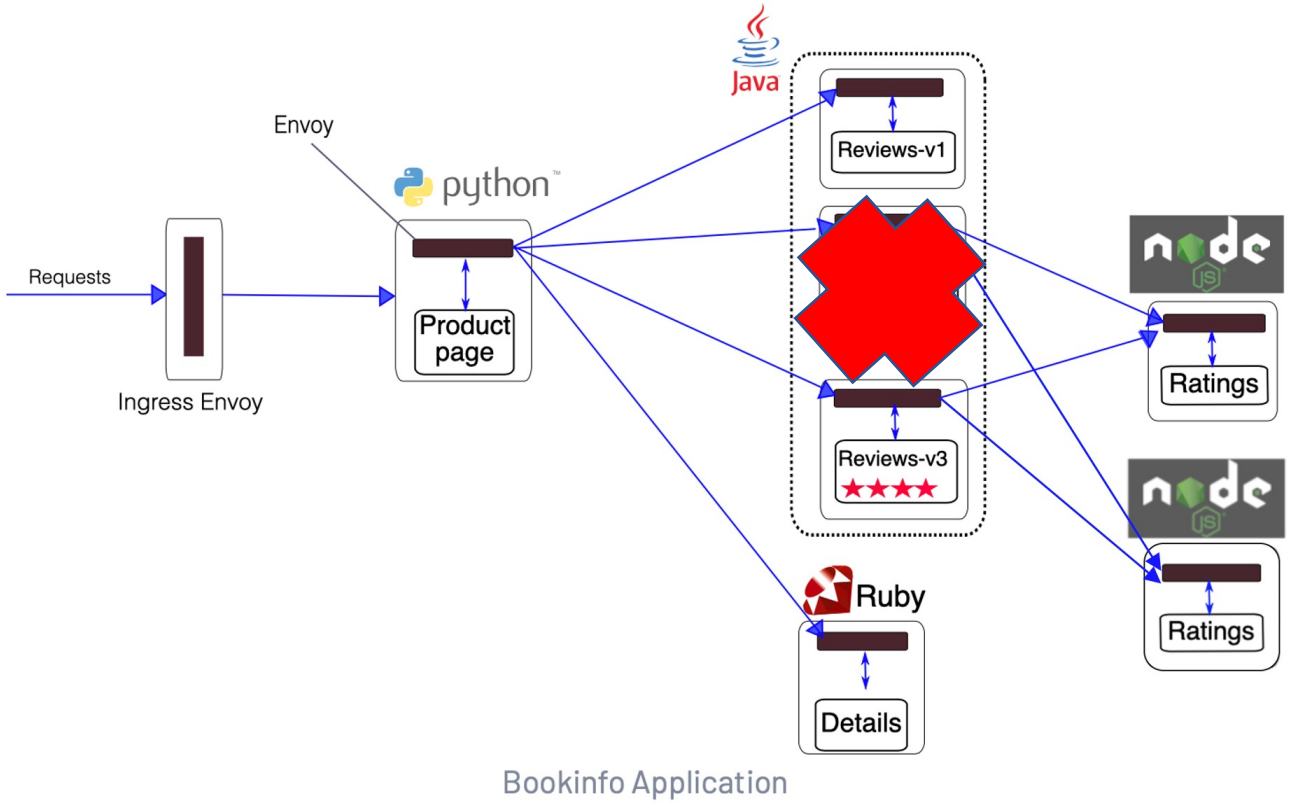
Model Parameters

$$w_{ij} = correlation(rt_{(i,\ j)},\ rt\_a(j))$$

$$AS_i = avg(w_i) * max\_correlation(metric,\ rt\_a(i))$$

Fault localization: Personalized PageRank

Anomalies Detection



Root cause:

[('reviews-v2', 0.392),
('productpage-v1', 0.333),
('ratings-v1', 0.163),
('ratings-v2', 0.096),
('reviews-v3', 0.014)]

Max correlated metric

rate_container_memory_working_set_bytes

Anomalies Detection

How to use it:

- Alerts
- Dynamic Thresholds
- Canaries Automation

What we do:

- Chaos Engineering

Predictive HPA

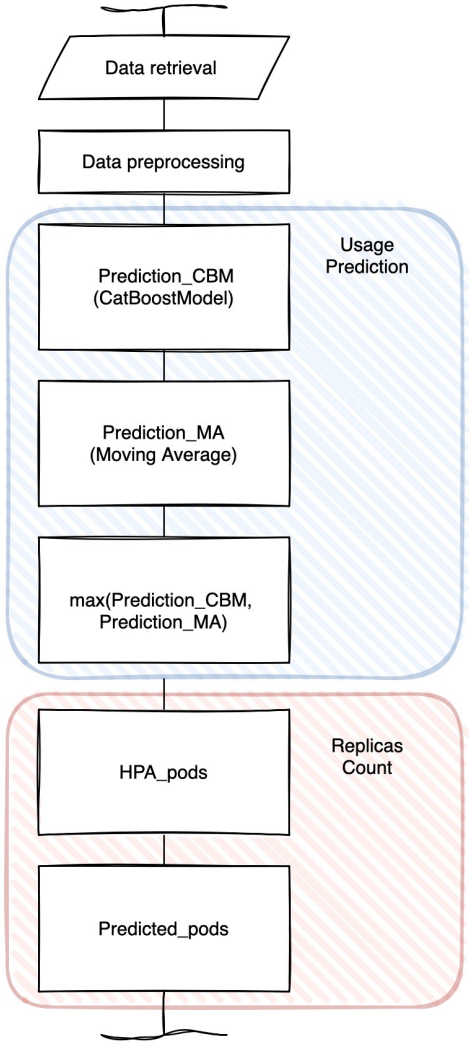
Assumption

- We can predict the load on the application in advance and trigger automatic horizontal scaling

Metrics list

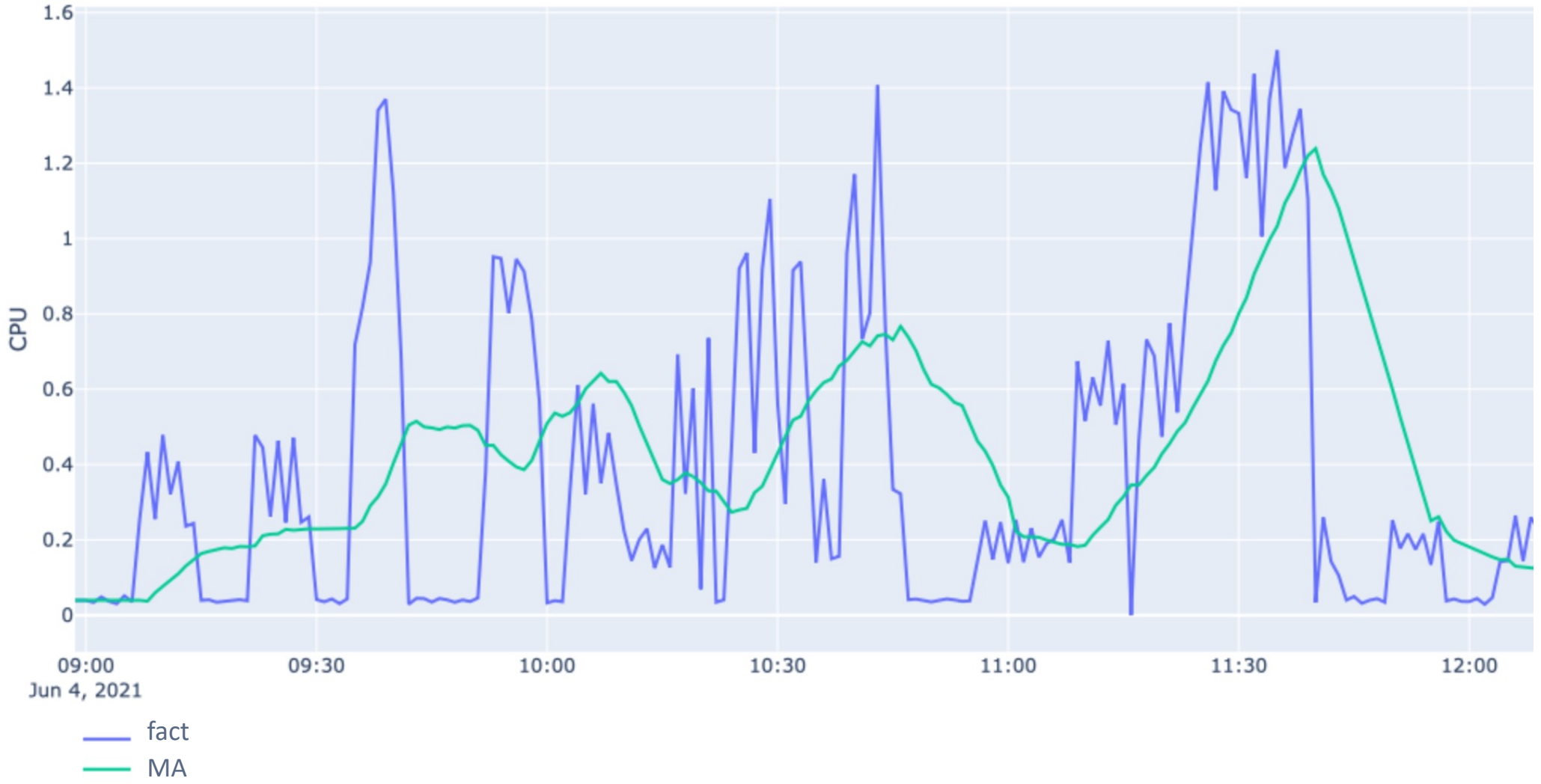
- container_sockets
- container_fs_inodes_total
- container_memory_cache
- container_memory_failcnt
- container_network_receive_errors_total
- container_network_transmit_errors_total
- container_network_transmit_packets_total
- container_fs_io_current
- container_fs_io_time_seconds_total
- container_fs_io_time_weighted_seconds_total
- container_fs_read_seconds_total
- container_fs_reads_merged_total
- container_spec_memory_swap_limit_bytes
- istio_tcp_sent_bytes_total
- container_cpu_usage_seconds_total
- container_memory_usage_bytes
- container_network_receive_bytes_total
- container_network_transmit_bytes_total
- container_memory_max_usage_bytes
- container_memory_failures_total
- container_memory_working_set_bytes
- container_memory_rss
- container_fs_usage_bytes
- container_cpu_system_seconds_total
- container_cpu_user_seconds_total
- container_fs_reads_total
- istio_request_duration_seconds_bucket

Predictive HPA

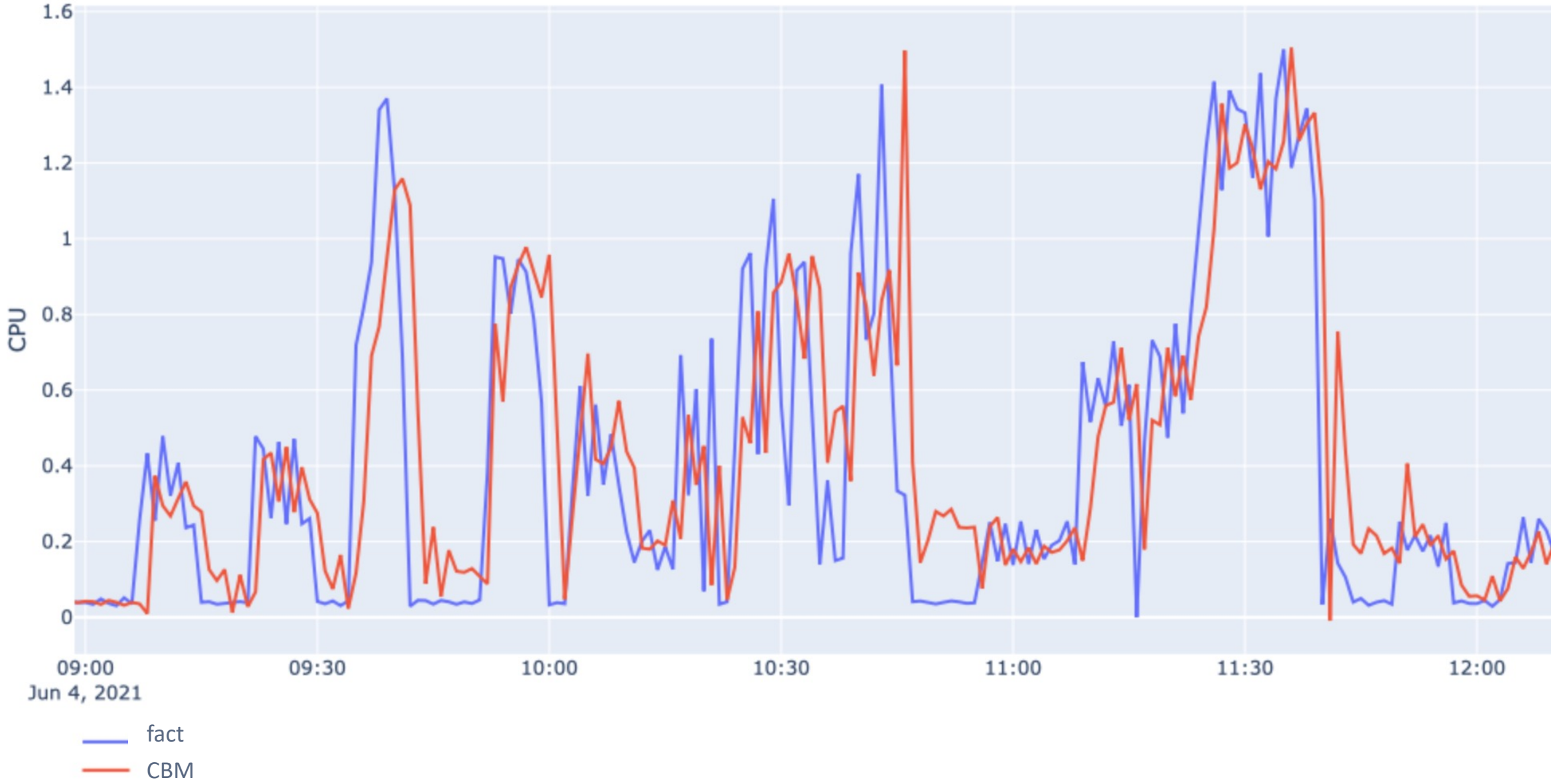


hour_0	...	hour_24	lag_1	lag_2	lag_3	...	lag_12
1	...	0	None	None	None	...	None
1	...	0	0	None	None	...	None
1	...	0	10	0	None	...	None
1	...	0	20	10	0	...	None
1	...	0	30	20	10	...	None

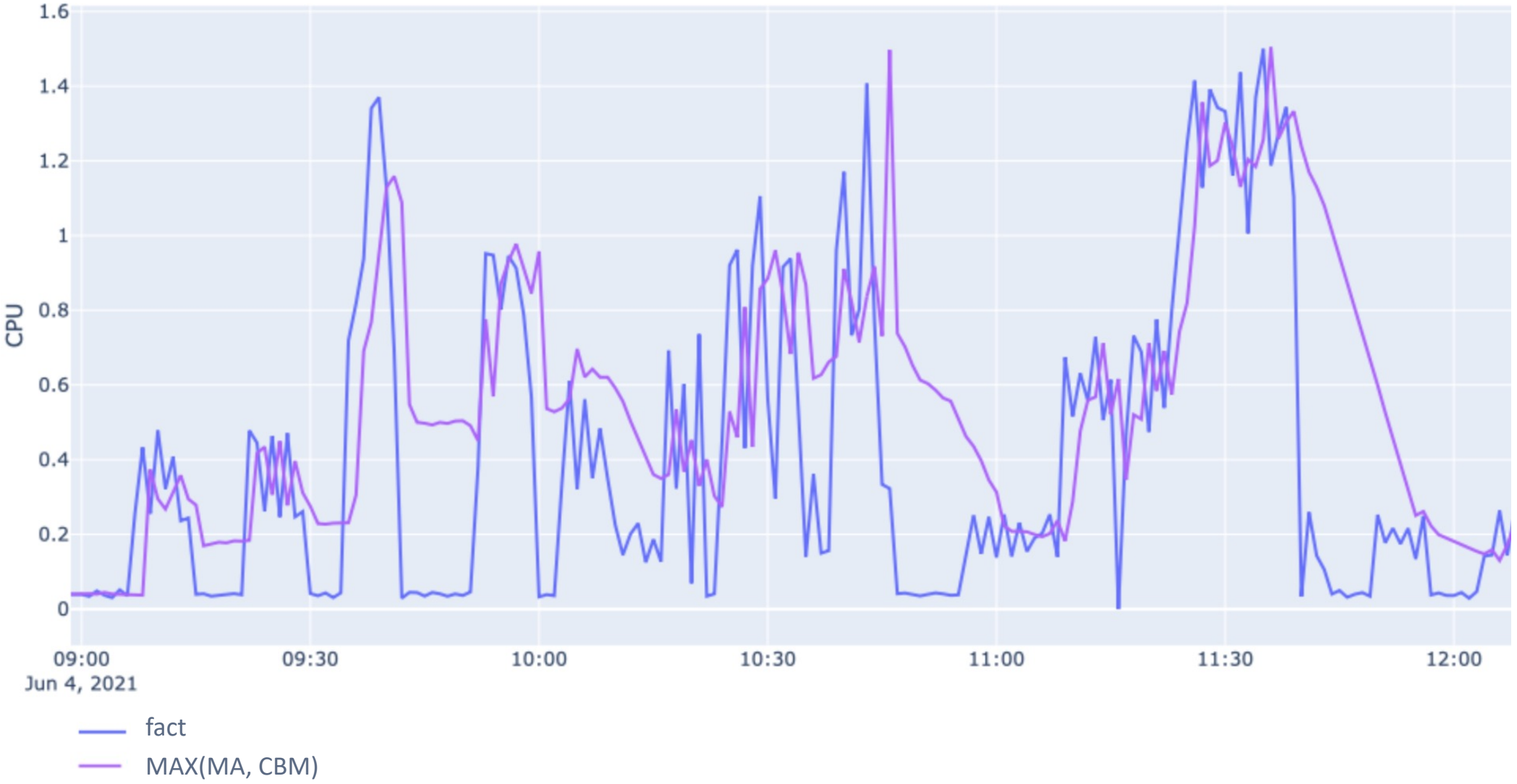
Predictive HPA



Predictive HPA



Predictive HPA



What's Next

- Migration from Recommendations to Operations
- Continuous Performance Monitoring
- Continuous Delivery for Models Lifecycle

Thank you!