中山大學 SUN YAT-SEN UNIVERSITY

ESEC/FSE 2023

Nezha: Interpretable Fine-Grained Root Causes Analysis for Microservices on Multi-Modal Observability Data



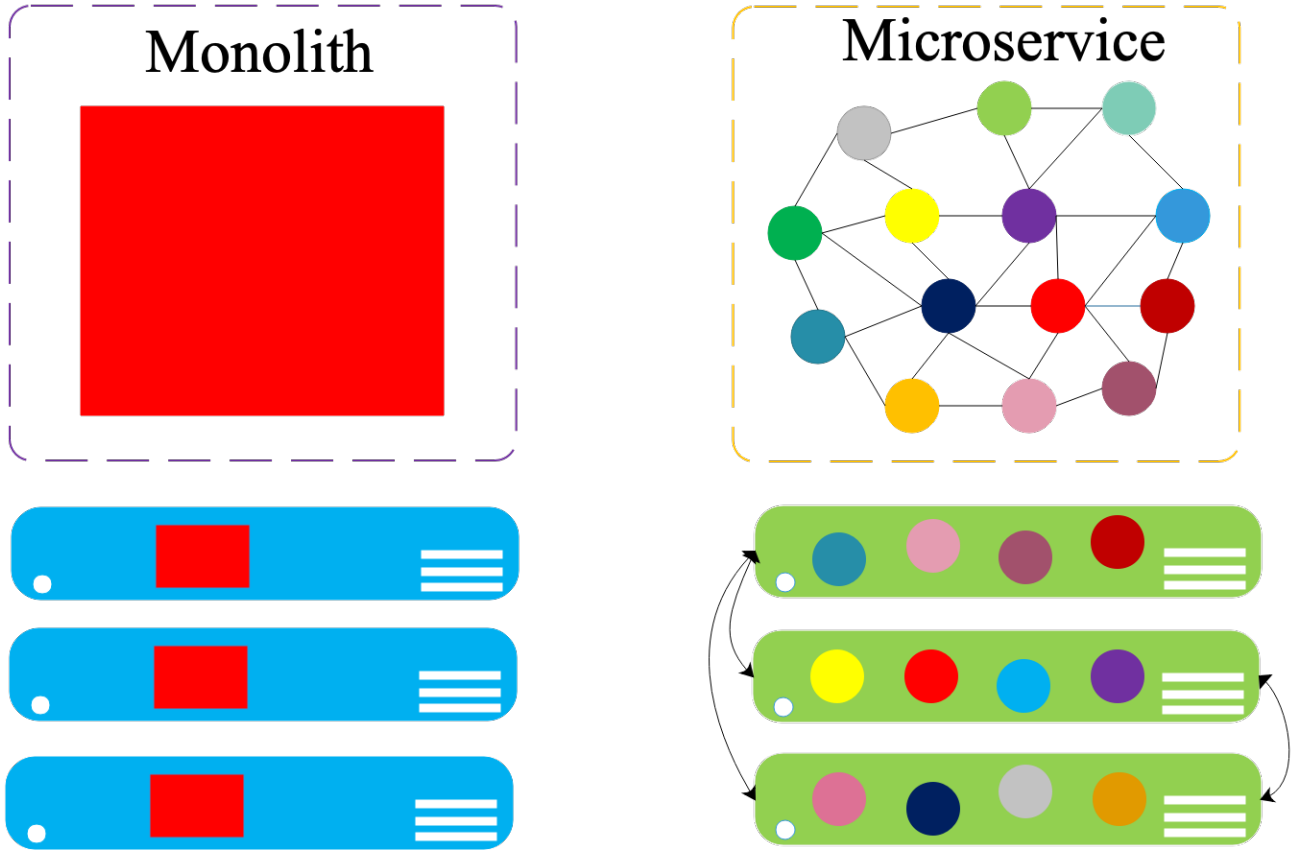



Guangba Yu Pengfei Chen Yufeng Li Hongyang Chen Xiaoyun Li Zibin Zheng

Sun Yat-sen University, Guangzhou, China

Introduction

- Microservice application consists of **many single-concerned, loosely-coupled** services
  - Fast development & deployment
  - On-demand provisioning, elasticity
  - Language/framework heterogeneity

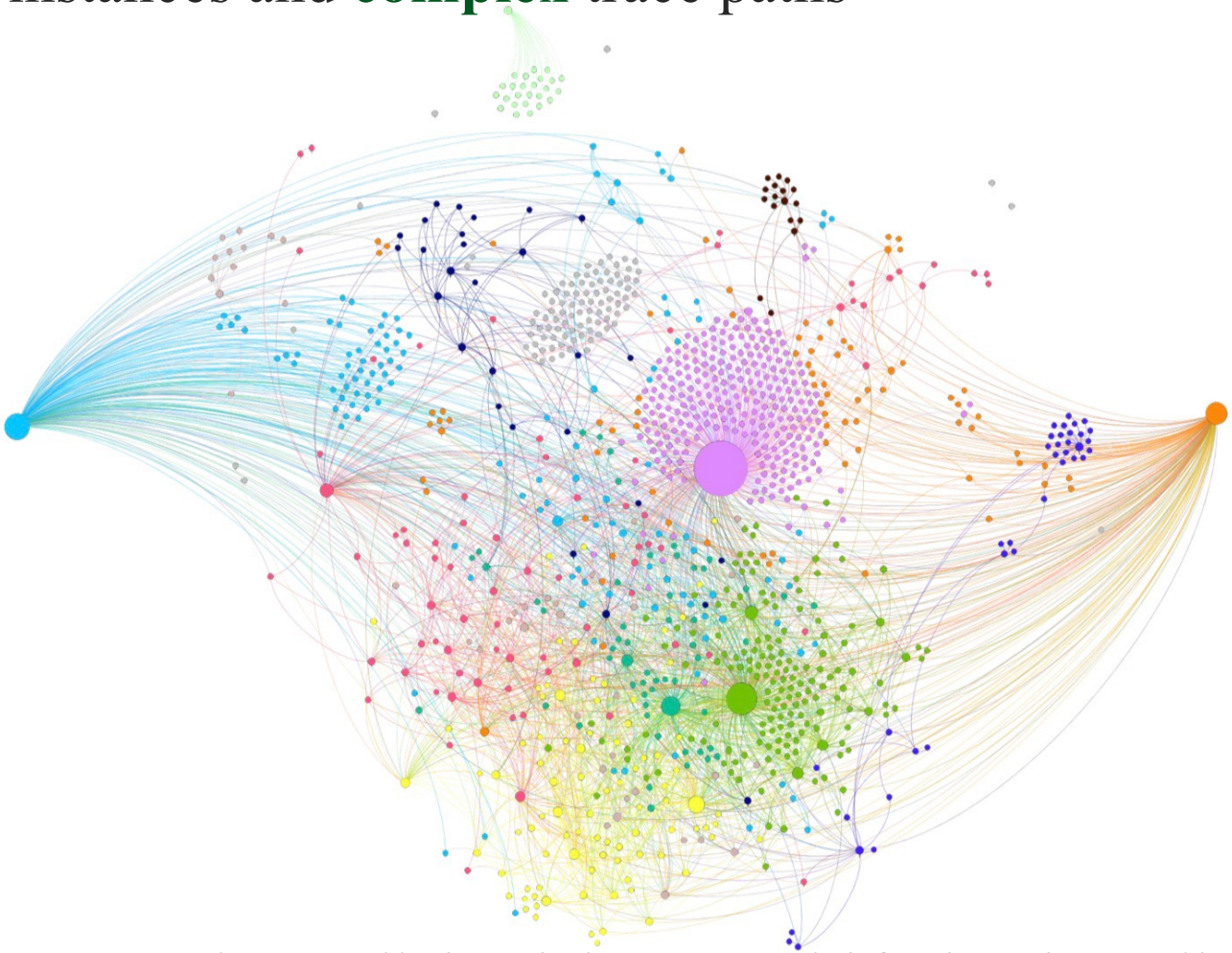


Introduction

- Failure is common in microservice systems

Introduction

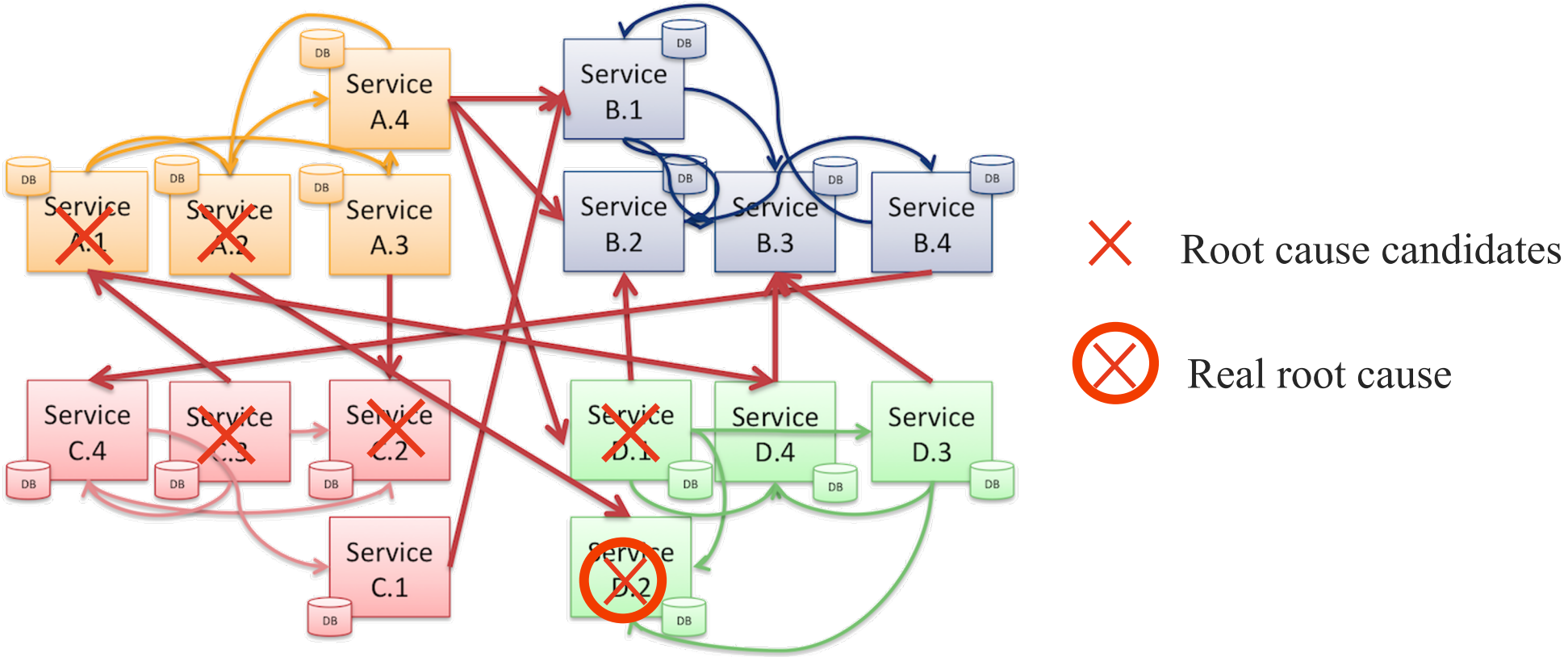
- Troubleshoot root causes in microservice application is time-consuming and error-prone
  - **Multiple** instances and **complex** trace paths



Dependency graph in Uber

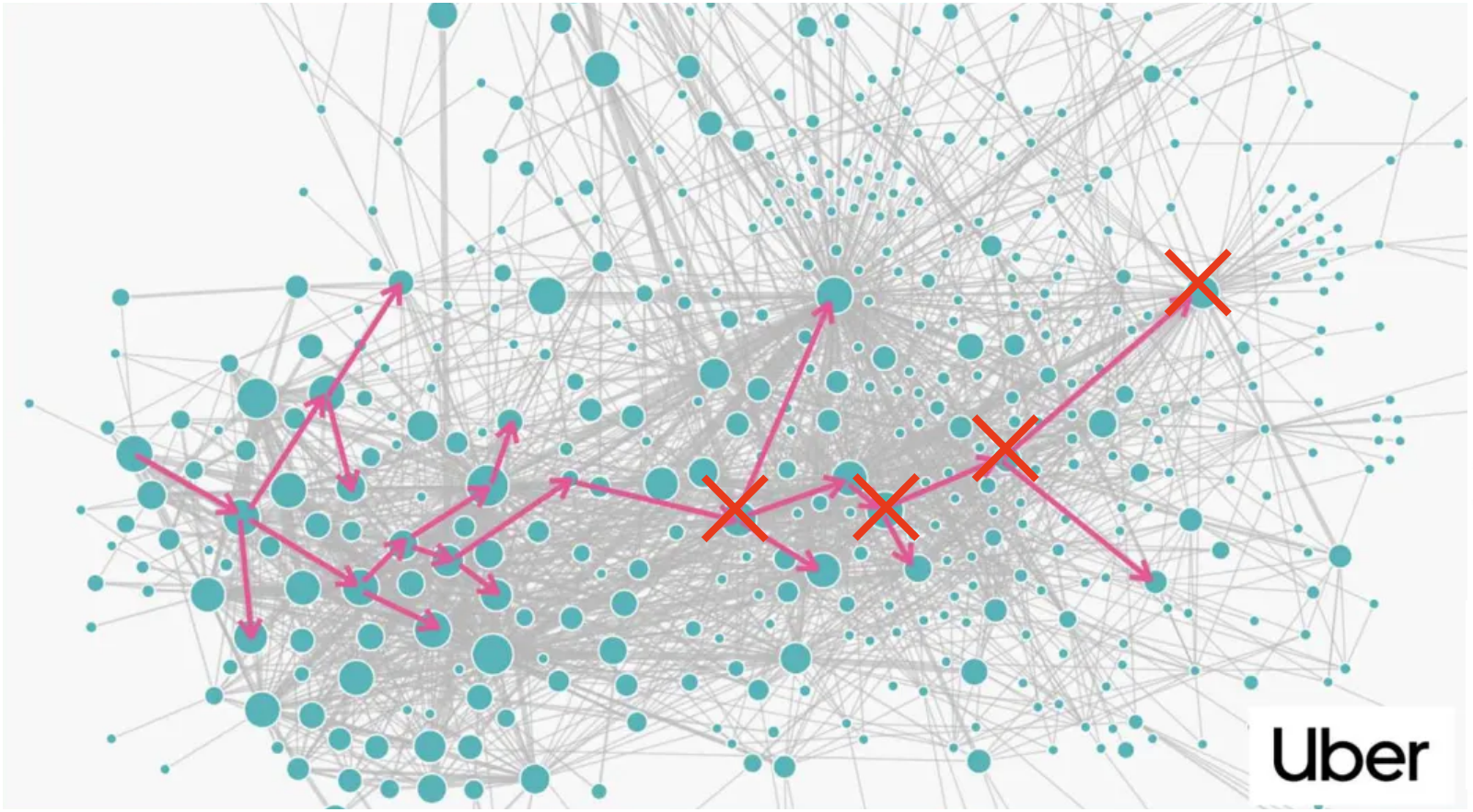
Introduction

- Root Cause Analysis in Microservice
  - Multiple instances and complex trace paths
  - Multiple root cause candidates caused by **anomaly propagation**



Introduction

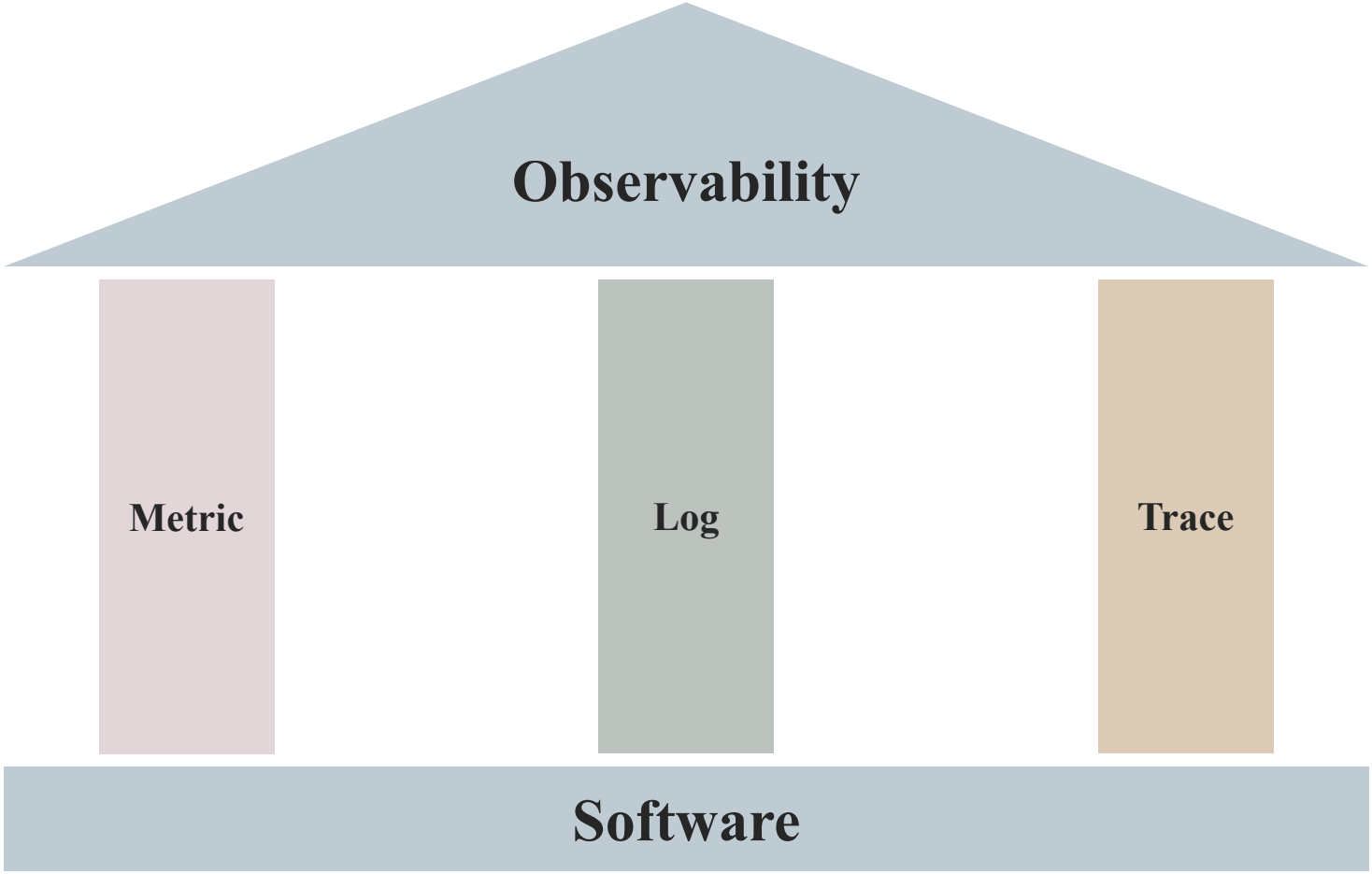
- Observability is crucial to localize root causes of microservice applications



What failed?
How should I fix them?

Introduction

- 3 pillars of observability: **logs, metrics and traces**

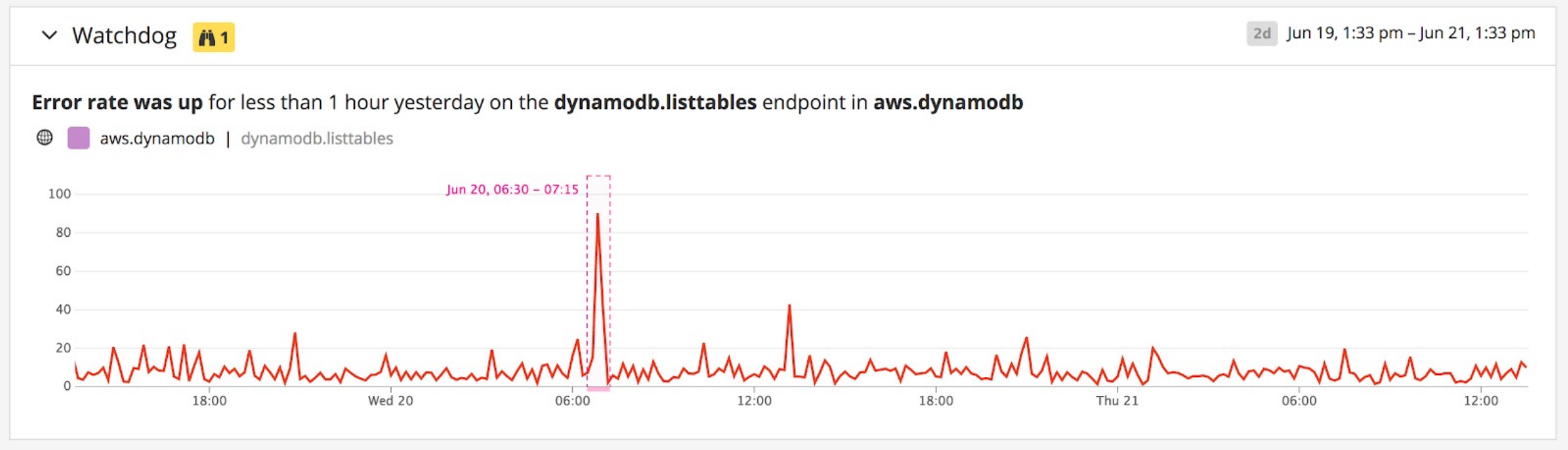


Introduction

- Metric are **aggregations** over a period of time of **numeric** data about infrastructure or application

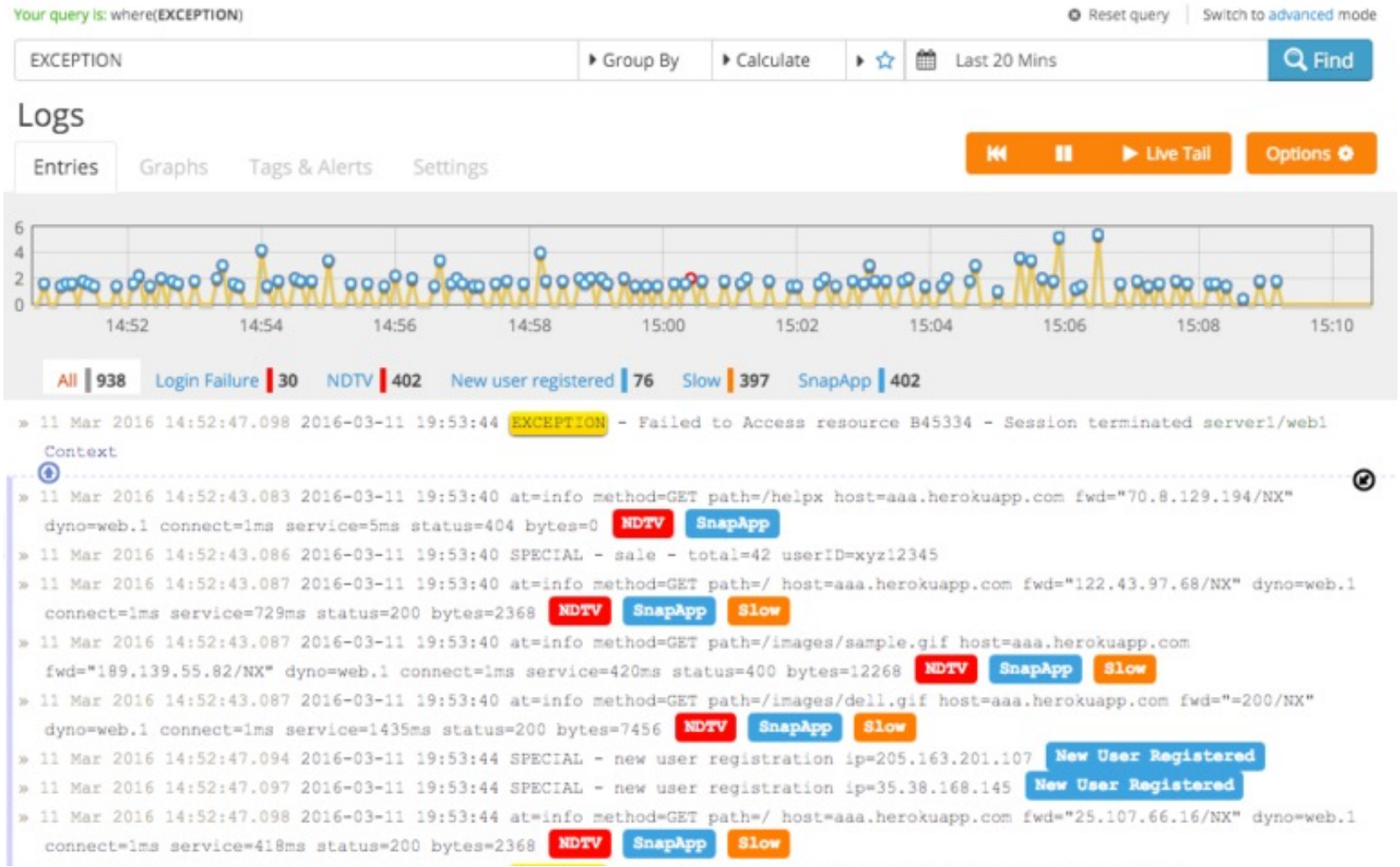
Nezha with Feng Huo Lun

Metrics : Error Rate of Request



Introduction

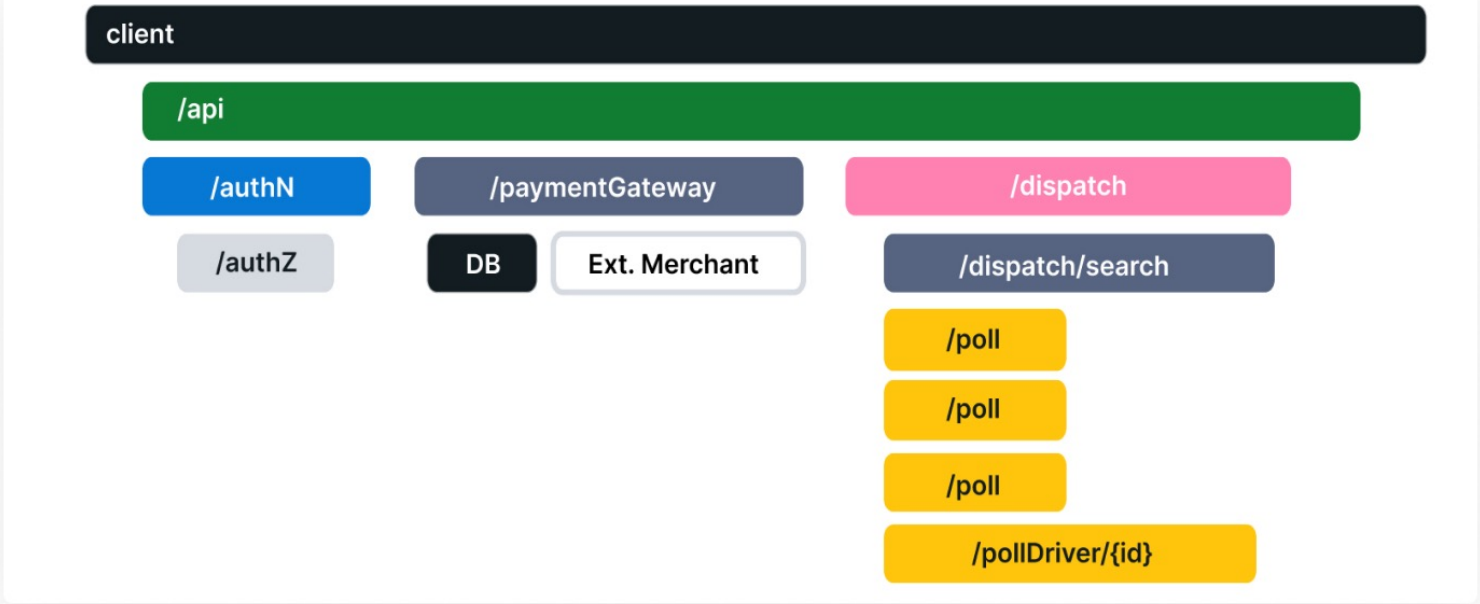
- Logs are **timestamped messages** emitted by services or other components



Nezha with Qian Kun Quan

Introduction

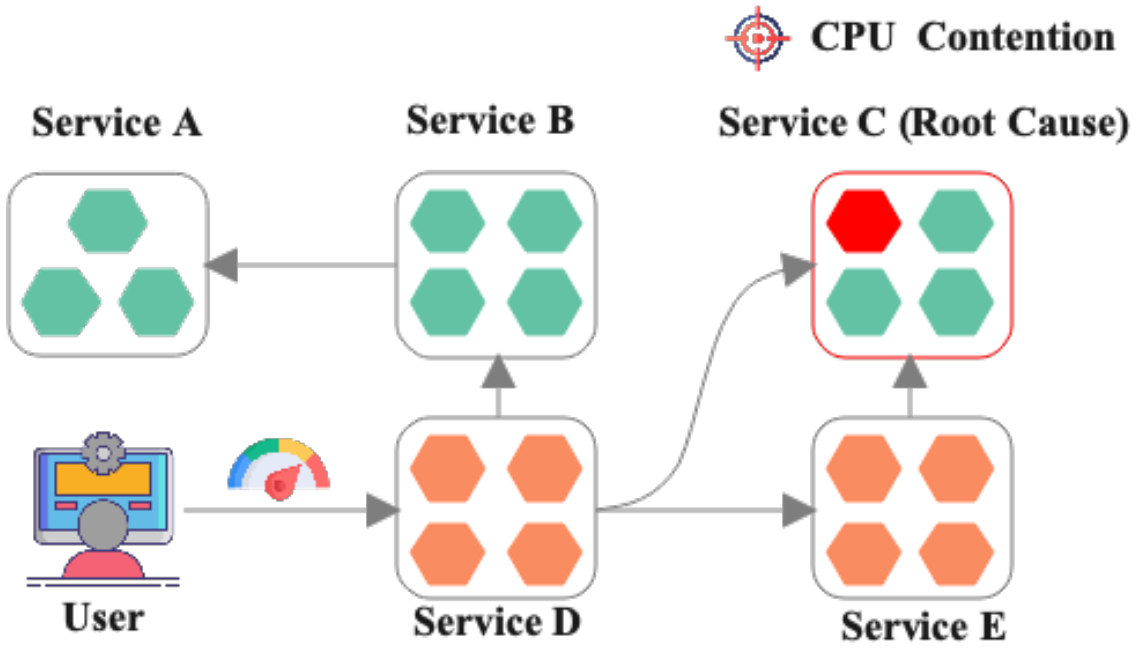
- Trace records the **paths** taken by requests as they **propagate** through multi-service
  - A trace is made of one or more spans
  - A span represents a unit of work or operation

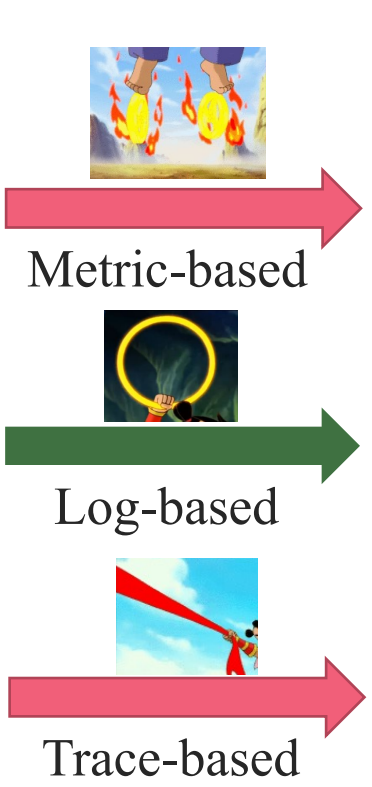


Nezha with
Hun Tian Ling

Motivation

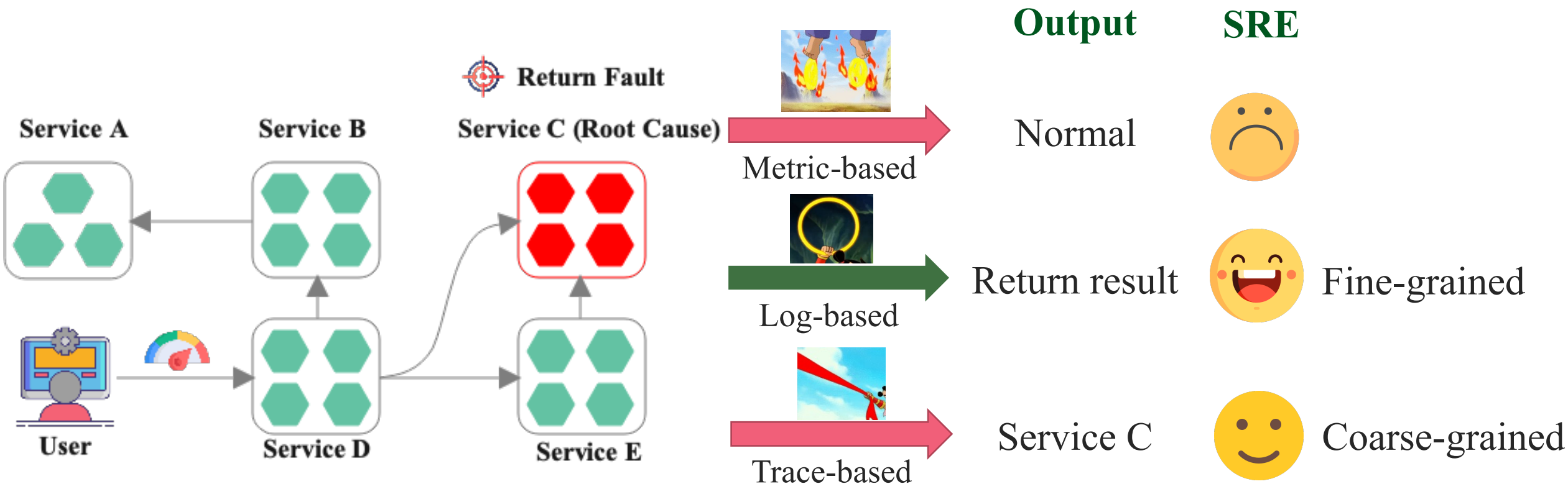
- Certain anomalies may not be apparent in some data sources
  - **CPU contention** is not obvious in **logs**





Motivation

- Certain anomalies may not be apparent in some data sources
  - **Return fault** is not obvious in **metrics**



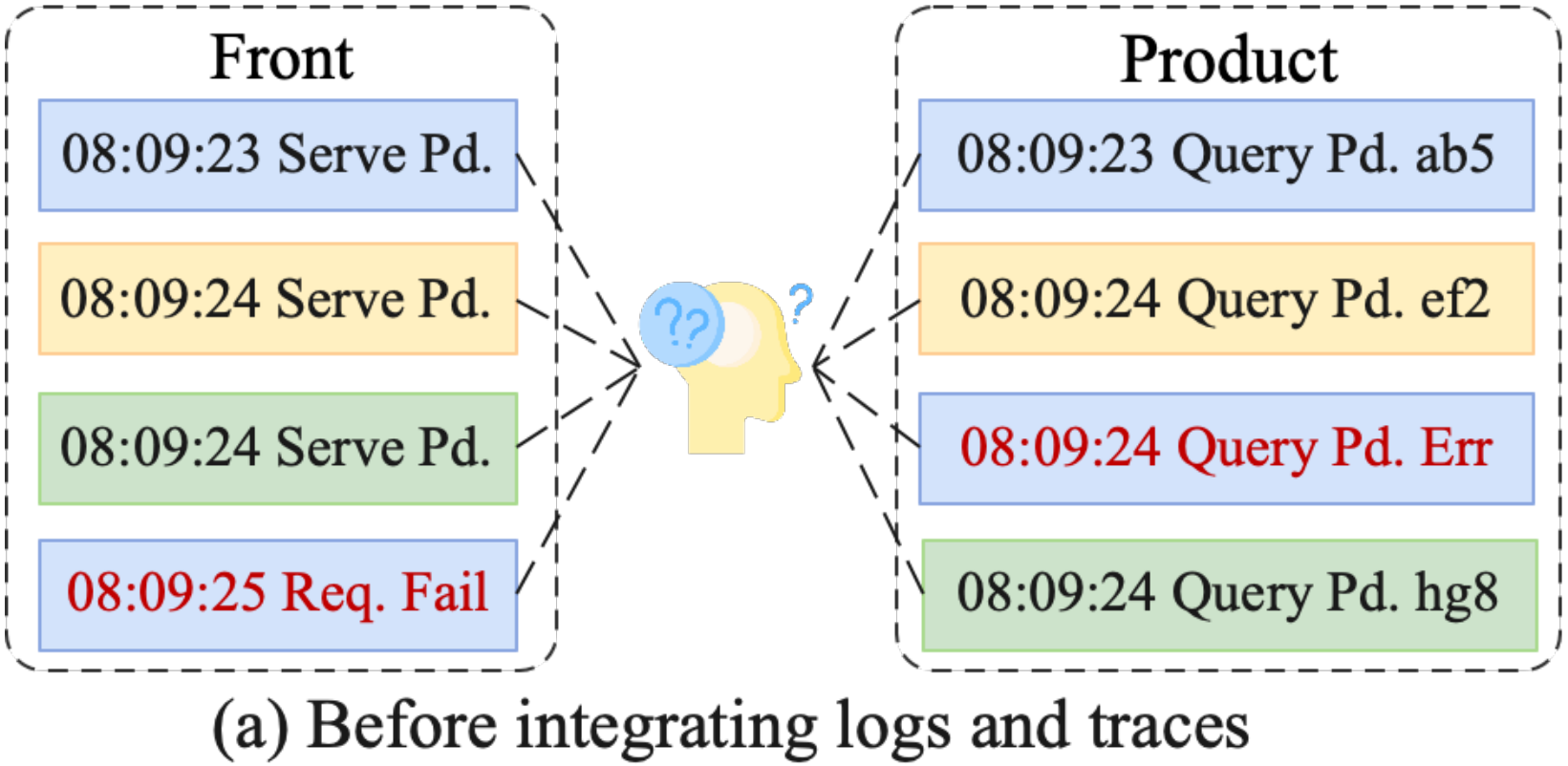
Motivation

- Certain anomalies may not be apparent in some data sources
  - CPU Contention is not obvious in logs
  - Return fault is not obvious in metrics

✓ **Motivation1: Enhancing RCA through the Integration of Metrics, Logs, and Traces**

Motivation

- Logs generated by a microservice can **interleave**, as it concurrently serves multiple requests

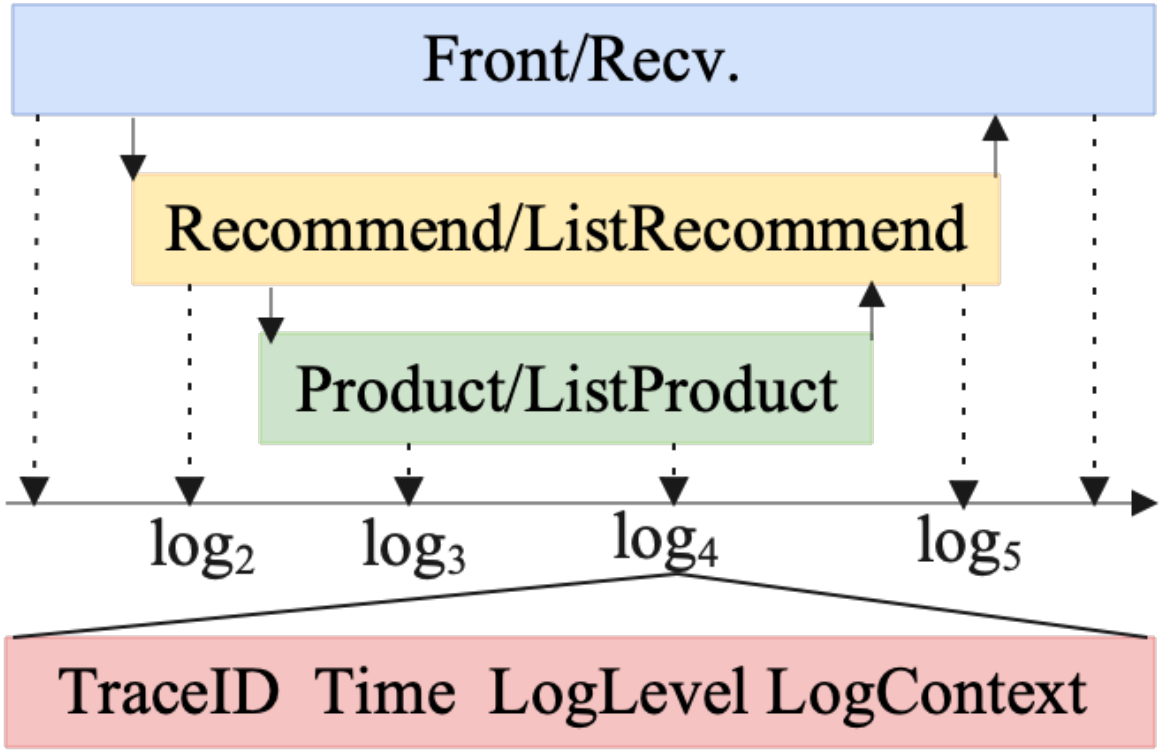


(a) Before integrating logs and traces

Which logs across multiple services are associated with the failed request?

Motivation

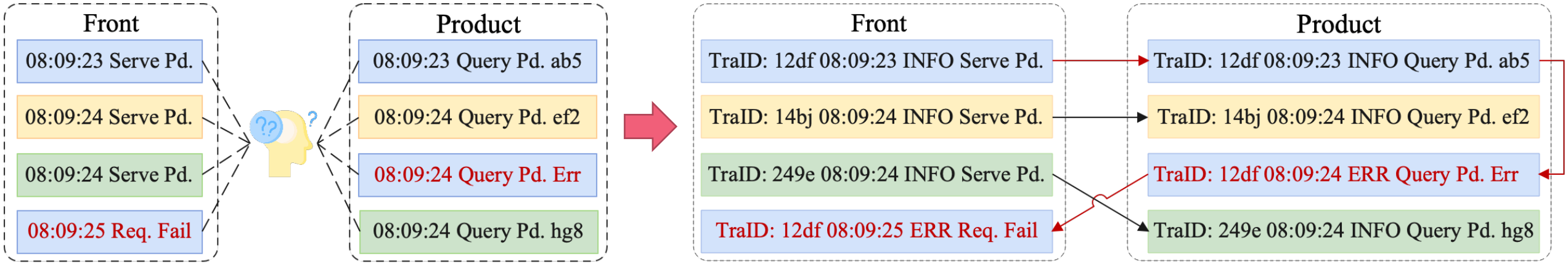
- Logs generated by a microservice can interleave, as it concurrently serves multiple requests
- **Inserting trace IDs** to logs to track the request-level information



(b) Integrating logs and traces

Motivation

- Logs generated by a microservice can interleave, as it concurrently serves multiple requests
- **Inserting trace IDs** to track the request-level descriptive information (e.g., logs)

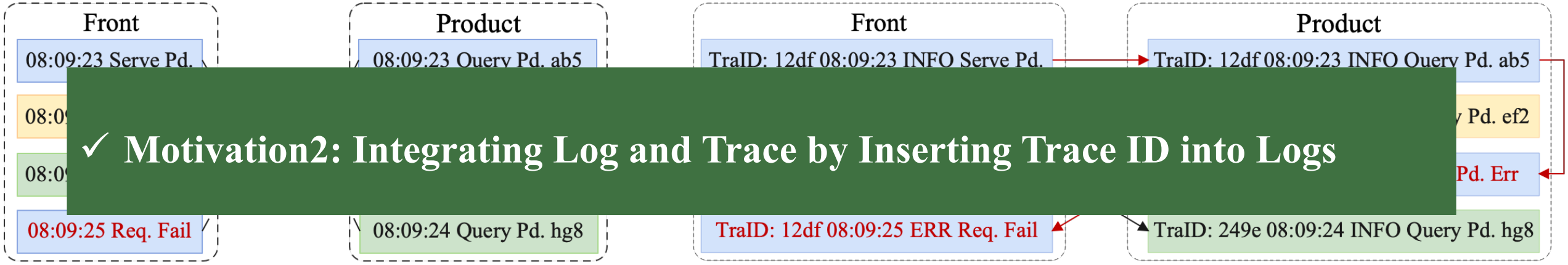


(a) Before integrating logs and traces

(c) After integrating logs and traces

Motivation

- Logs generated by a microservice can interleave, as it concurrently serves multiple requests
- Inserting trace IDs to track the request-level descriptive information (e.g., logs)



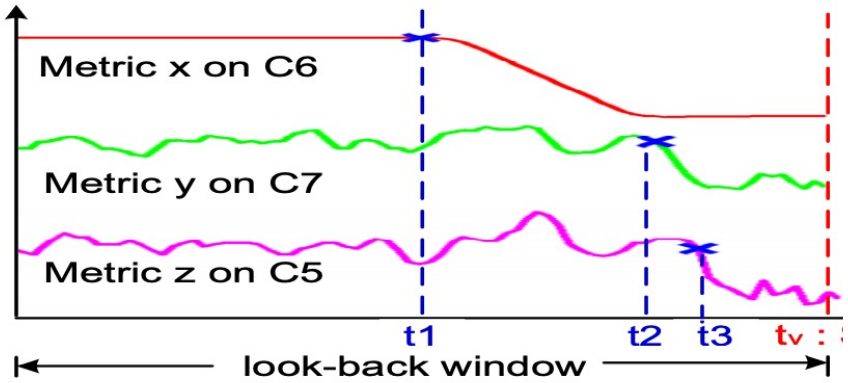
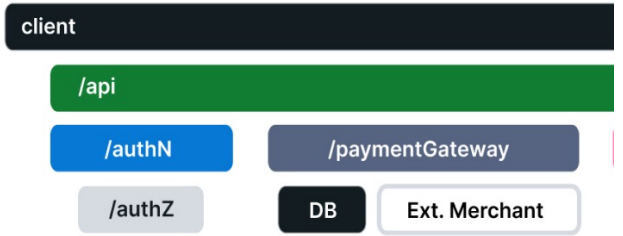
✓ **Motivation2: Integrating Log and Trace by Inserting Trace ID into Logs**

(a) Before integrating logs and traces

(c) After integrating logs an

Motivation

- Metrics, Log, Trace are **heterogeneous observability data**

Metrics	Log	Trace
		
Numerical values	Text	Semi-structured text

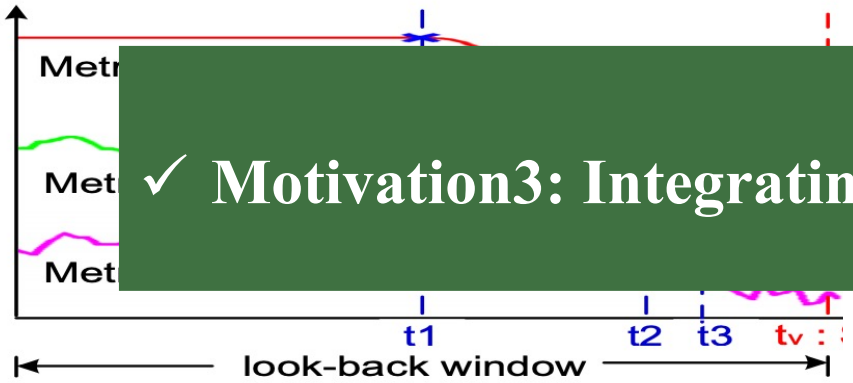
Motivation

- Metrics, Log, Trace are heterogeneous observability data

Metrics

Log

Trace



✓ Motivation3: Integrating Multi-modal Data in a Unified Representation

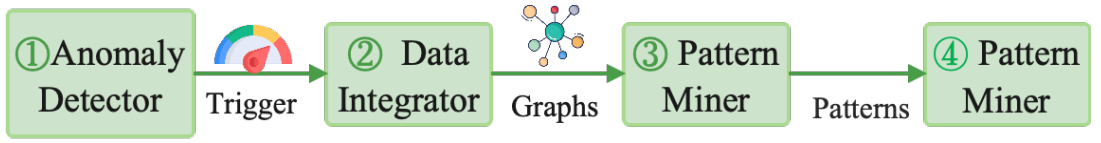
10:00:02 INFO AddStoredBlock

Numerical values

Textual records

Semi-structured text

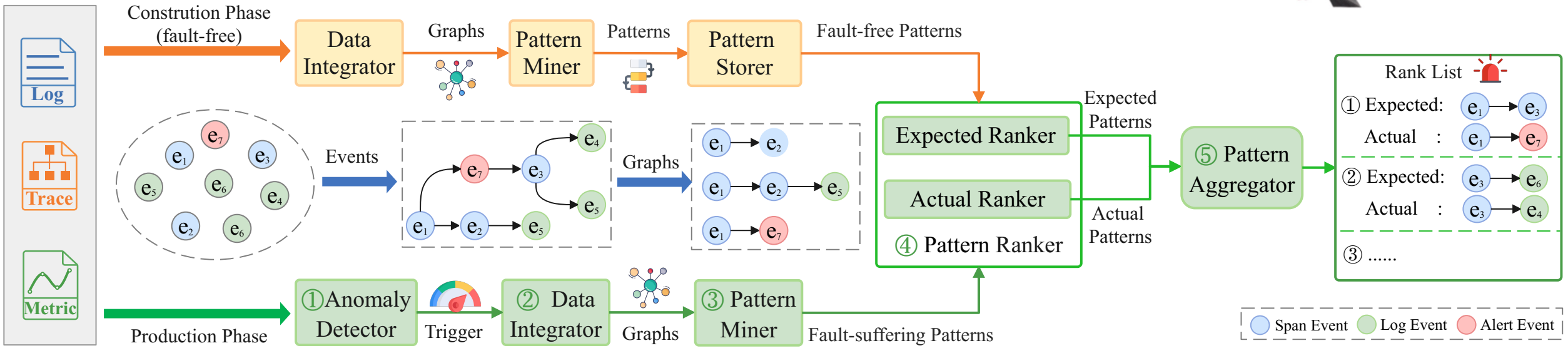
Nezha Approach



- Nezha: an **unsupervised fine-grained** RCA approach by incorporative analysis of multi-modal data **in an interpretable manner**

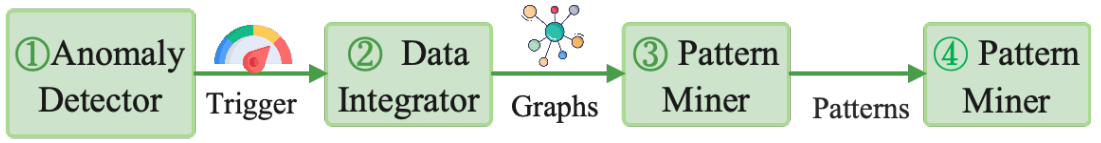
① Trigger by existing anomaly detection approaches

Fault-free Construction Phase



Fault-suffering Production Phase

Nezha Approach



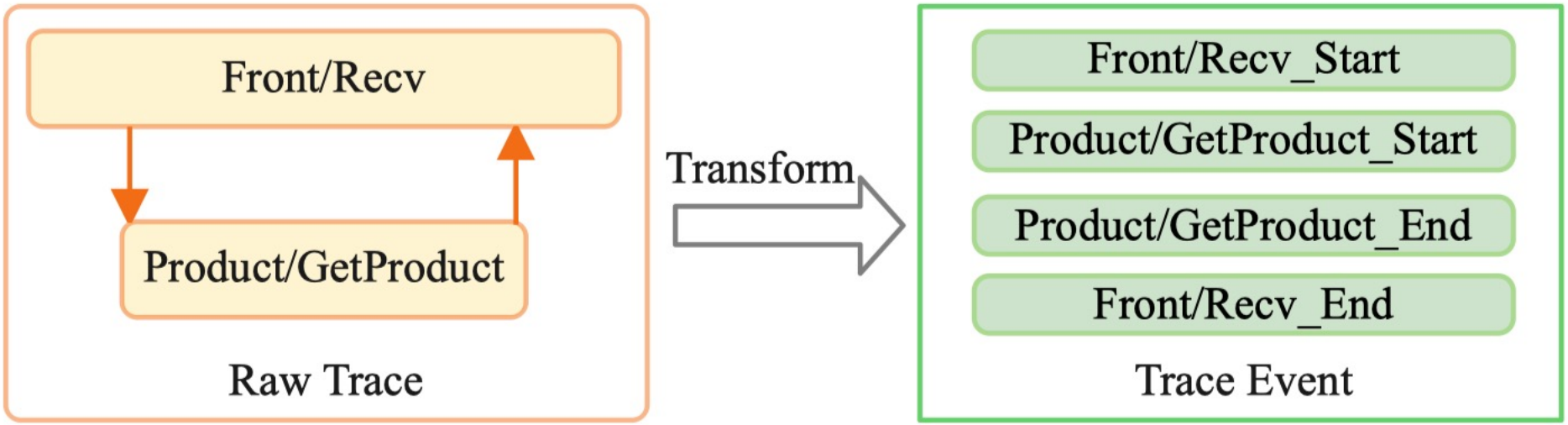
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

  ② Integrate multi-modal data into **event graph**

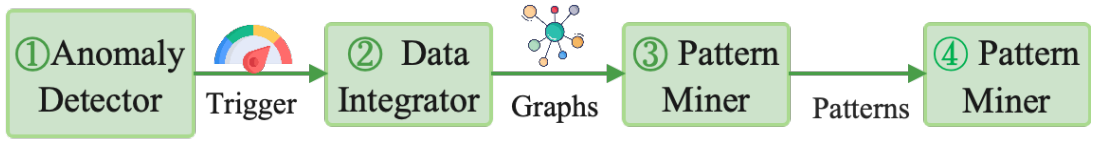
  I. Transform metrics as **metric alert events** by anomaly detection

  II. Transform logs as **log events** by log parsing

  III. Transform traces as **traces events**



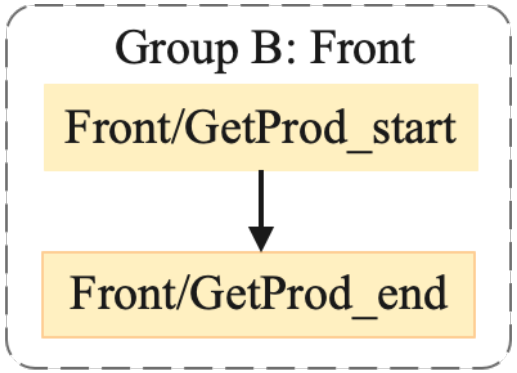
Nezha Approach

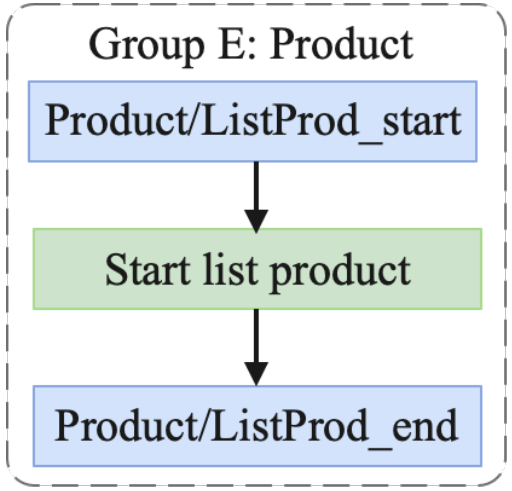


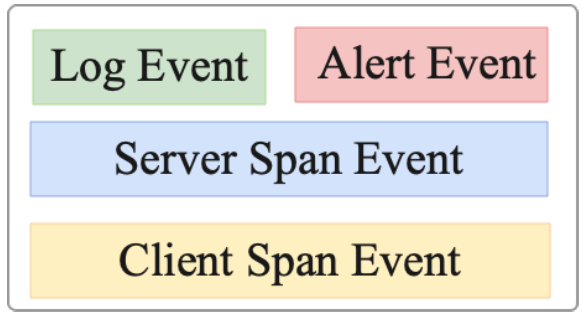
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

  ② Integrate multi-modal data into event graph based on timestamps and trace ID

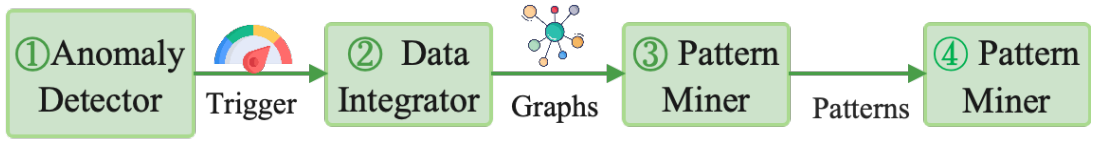
  IV. Order events in the **same span** based on **timestamps**







Nezha Approach

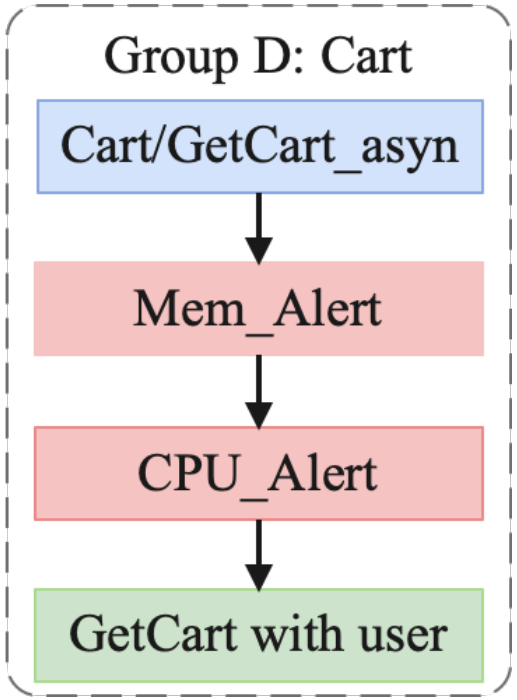


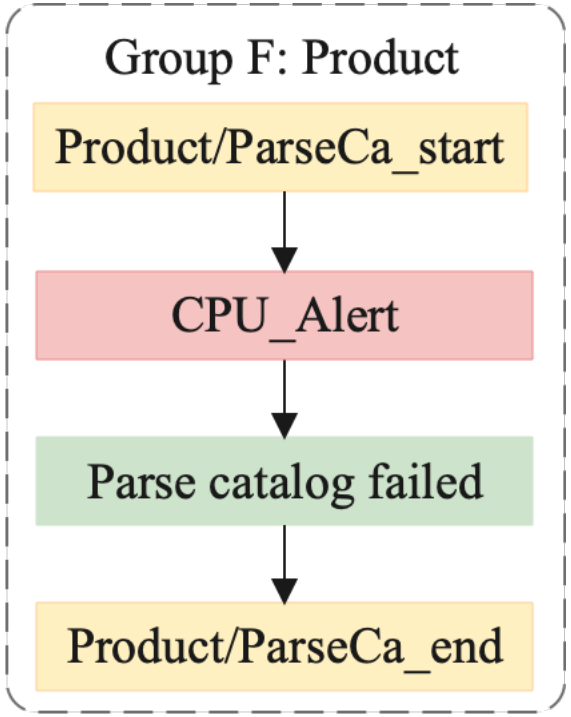
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

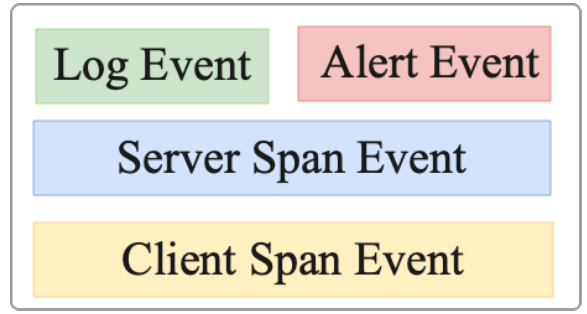
  ② Integrate multi-modal data into event graph based on timestamps and trace ID

  IV. Order events in the same span based on timestamps

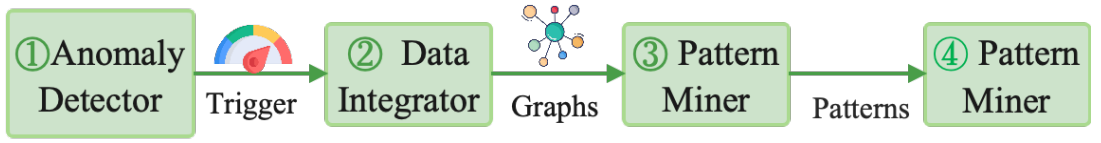
  V. **Insert metric events** to event groups







Nezha Approach



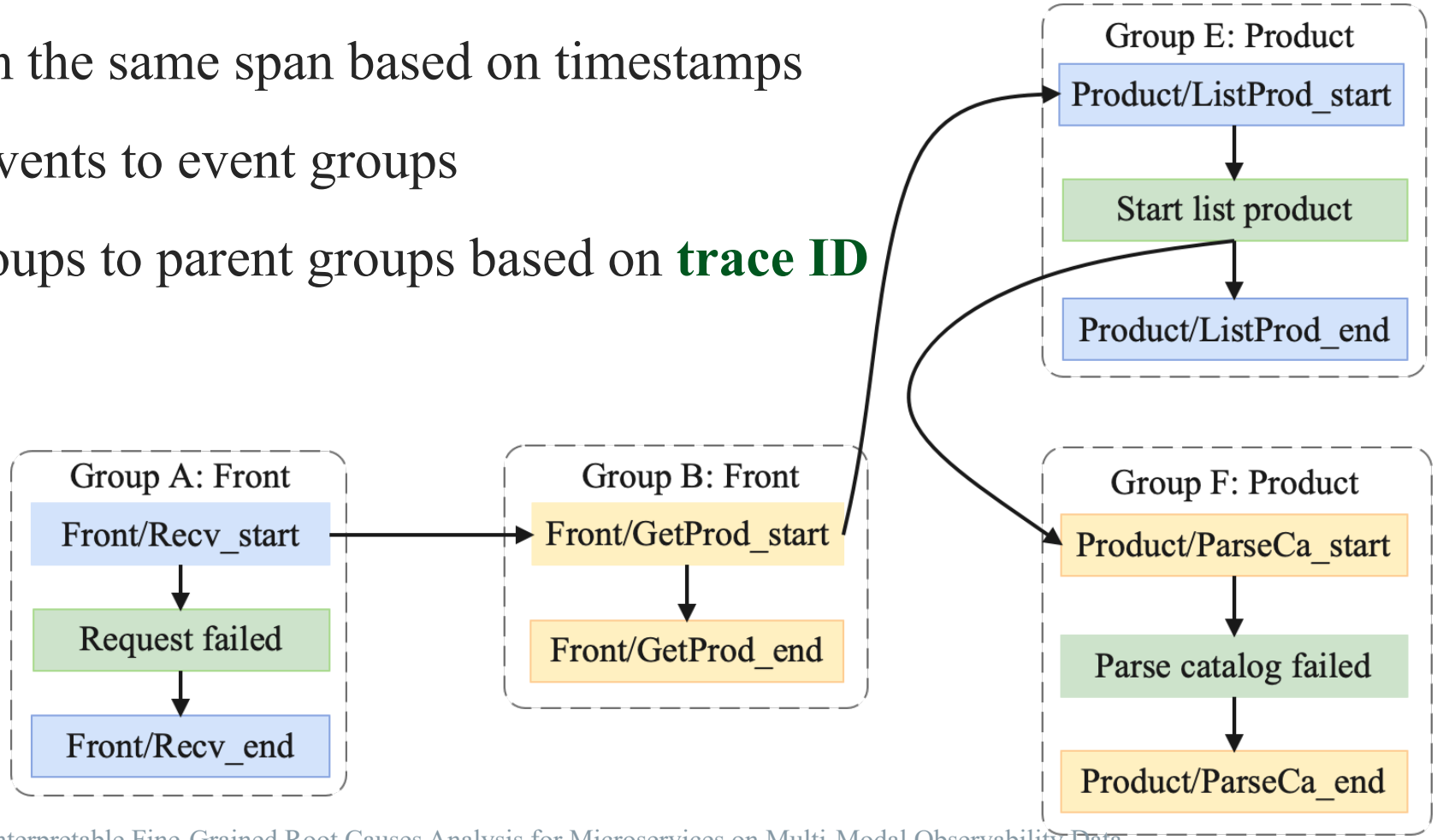
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

  ② Integrate multi-modal data into event graph based on timestamps and trace ID

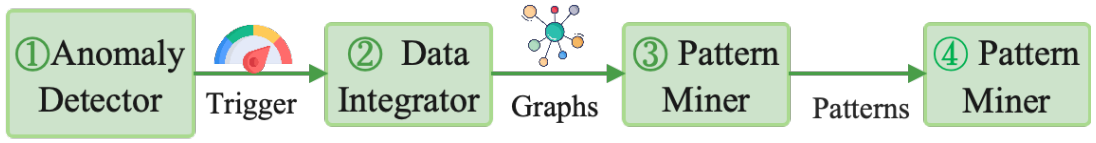
  IV. Order events in the same span based on timestamps

  V. Insert metric events to event groups

  VI. Insert child groups to parent groups based on **trace ID**



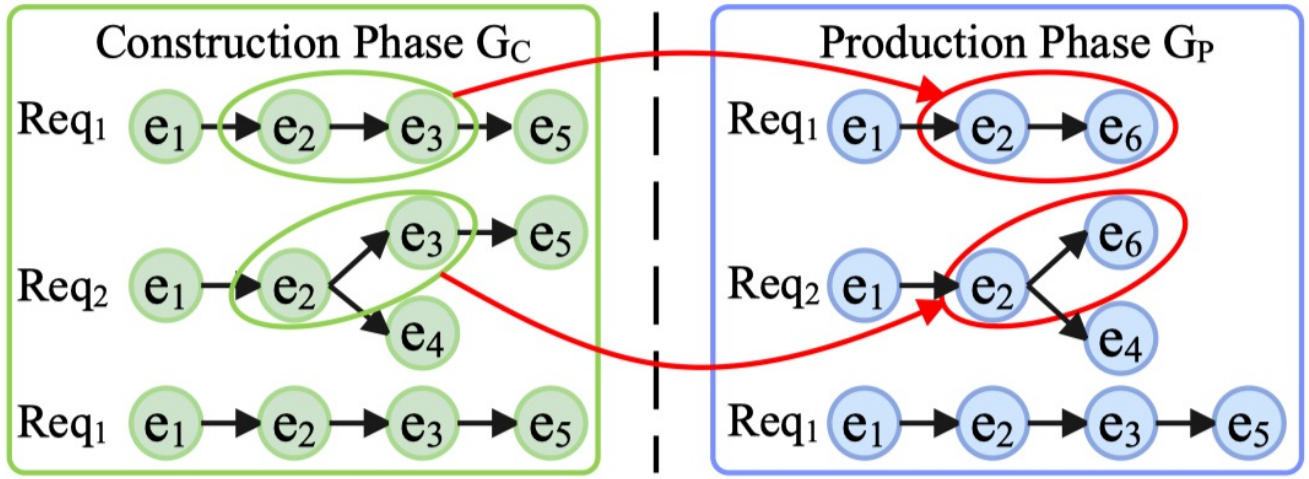
Nezha Approach



- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

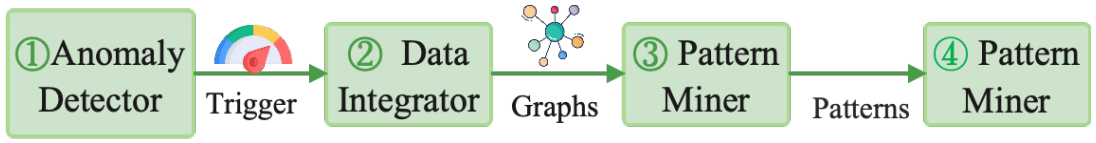
  ③ Mine frequent patterns from event graphs

  I. Count the **occurrences** of each pattern to calculate the support of each pattern



Pattern	Support	
	S_C	S_P
$e_1 \rightarrow e_2$	3	3
$e_2 \rightarrow e_3$	3	1
$e_2 \rightarrow e_4$	1	1
$e_3 \rightarrow e_5$	3	1
$e_2 \rightarrow e_6$	0	2

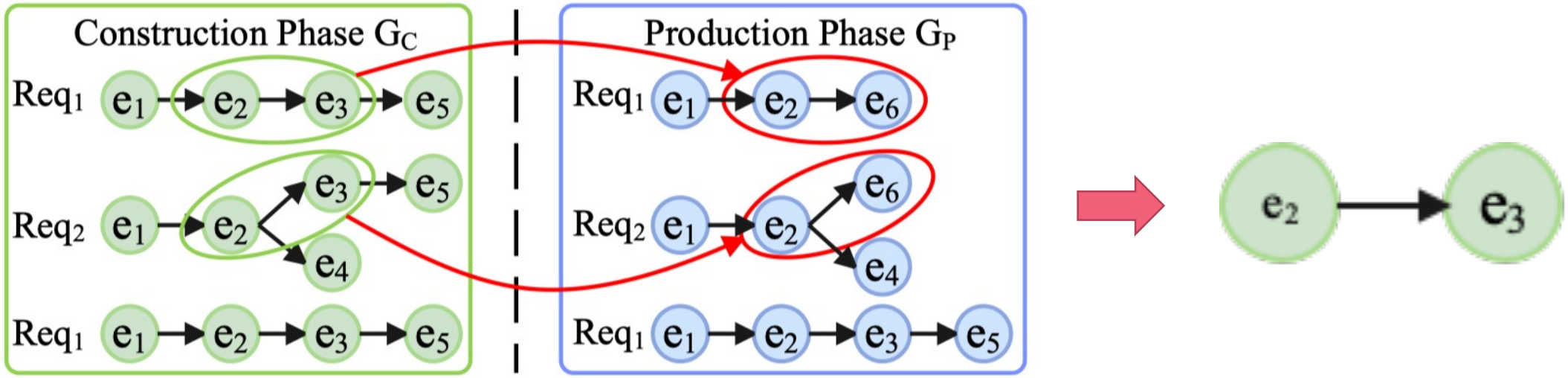
Nezha Approach



- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

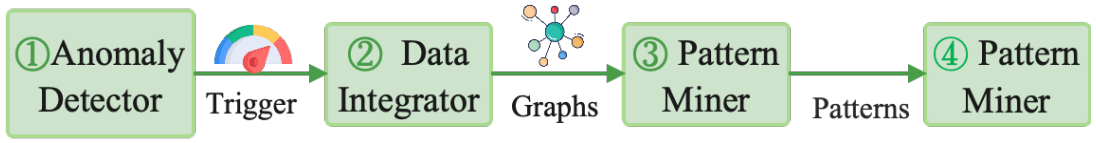
  ④ Rank suspicious patterns by comparing patterns between fault-free and fault-suffering

  I. Expected Pattern: identify which patterns **do not follow expected execution** paths



✓ **Core Idea: Rank event patterns that occur multiple times in the fault-free phase but rarely in the fault-suffering phase above other event patterns**

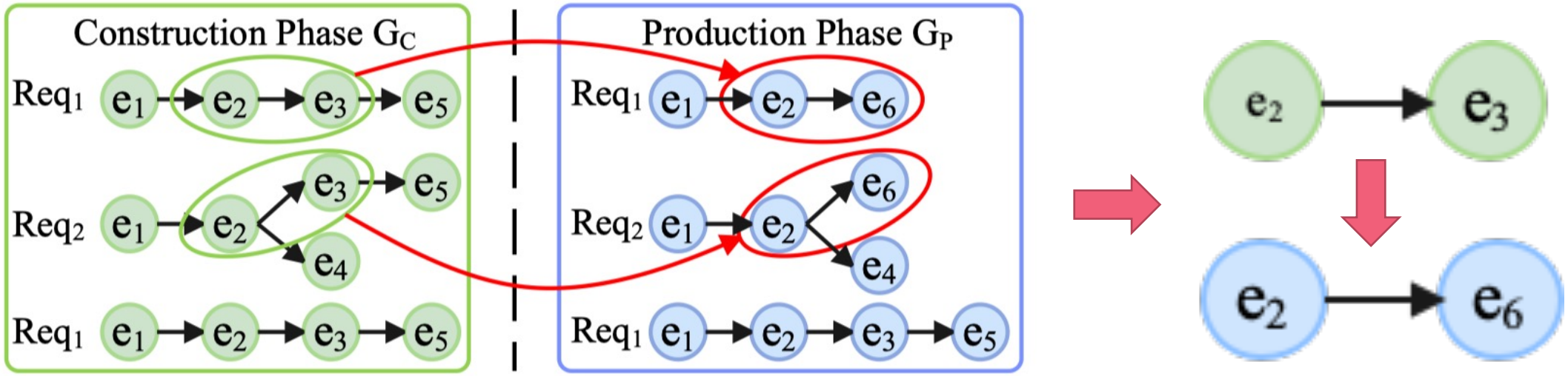
Nezha Approach



- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

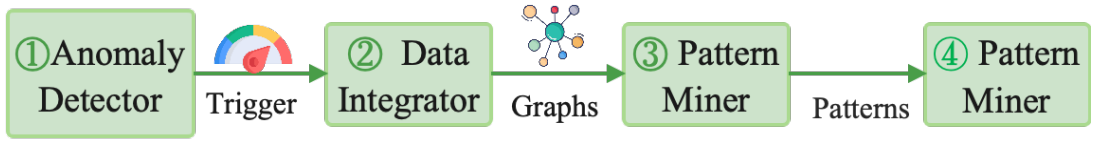
④ Rank suspicious patterns by comparing patterns between fault-free and fault-suffering

II. Actual Pattern: **how expected patterns change** in the actual fault-suffering phase



✓ **Core Idea: Rank event patterns that occur multiple times in the fault-suffering phase but rarely in the fault-free phase above other event patterns.**

Nezha Approach



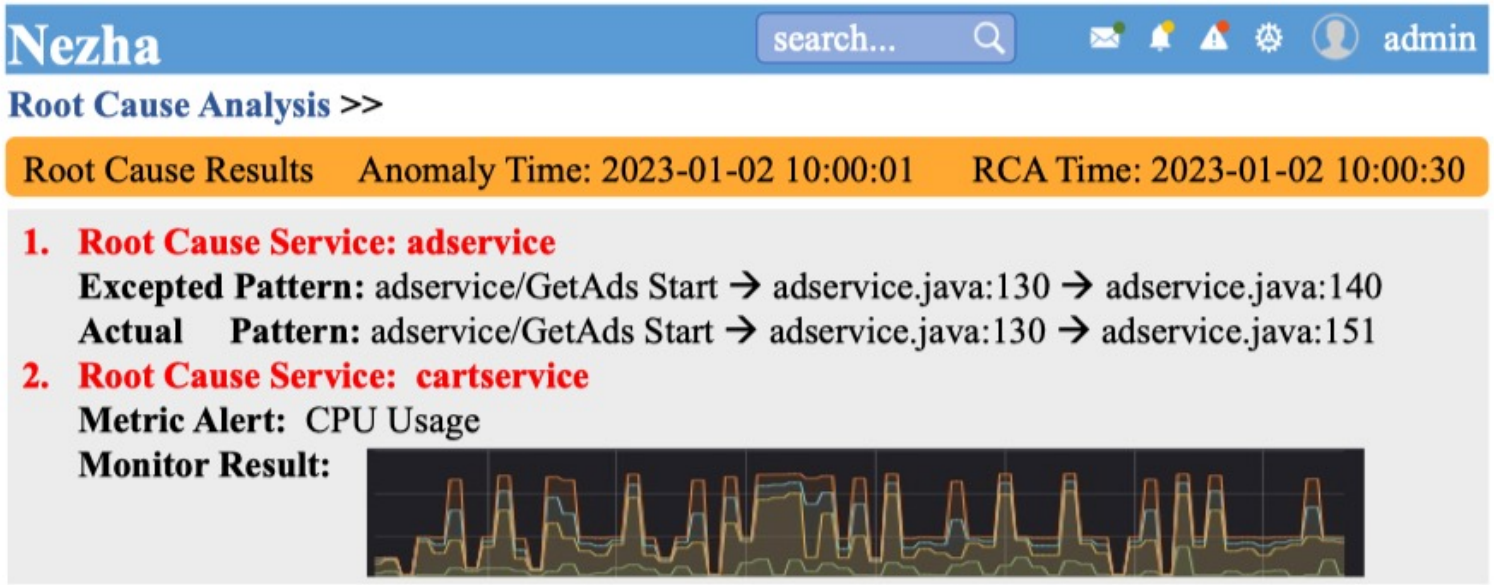
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

  ④ Rank suspicious patterns by comparing patterns between fault-free and fault-suffering

  I. Expected Pattern: identify which patterns do not follow expected execution paths

  II. Actual Pattern: how expected patterns change in the actual fault-suffering phase

  III. Correlate expected patterns with actual patterns to provide the complete fault scene



Experiment Set up

- Benchmark
  - OnlineBoutique[1]
  - TrainTicket[2]

```
log.error('Req Failed')                                          Source Code
```

```
trace_id = '{trace:032x}'.format(trace=ctx.trace_id)             Modified Code
span_id = '{span:016x}'.format(span=ctx.span_id)
log.error('Trace_id=%s Span_id=%s Req Failed', trace_id, span_id)
```

Insert trace ID into logs

- Fault
  - CPU contention
  - Network jam
  - Error return
  - Exception code

	OnlineBoutique	TrainTicket
CPU Contention	20	10
Network jam	22	10
Error return	8	13
Exception code	6	12
Total	56	45

1. https://github.com/IntelligentDDS/Augmented-OnlineBoutique
2. https://github.com/IntelligentDDS/Augmented-TrainTicket

Experiment Result

- Root cause results at service level

Table 3: Comparison of baselines at service level.

Approach	OnlineBoutique			TrainTicket		
	AS@1	*AS*@3	*AS*@5	*AS*@1	*AS*@3	*AS*@5
MicroScope	12.5	41.07	55.35	17.78	26.67	35.56
MicroRCA	16.07	62.5	92.75	20.00	31.11	44.44
SBLD	19.64	23.21	25.00	15.56	22.22	24.44
LogFaultFlagger	19.64	21.42	23.21	17.78	24.44	24.44
MicroRank	41.07	48.21	62.5	15.56	24.44	35.56
TraceAnomaly	30.35	33.92	48.21	13.33	28.89	33.33
PDiagnose	41.07	73.21	82.14	8.89	13.33	22.22
Nezha w/o $\mathcal{ML}$	14.28	17.85	17.85	6.67	8.89	11.11
Nezha w/o $\mathcal{M}$	26.78	33.92	35.71	55.56	62.22	68.89
Nezha w/o $\mathcal{L}$	64.28	64.28	64.28	42.22	44.44	44.44
Nezha	**92.86**	**96.43**	**96.43**	**86.67**	**97.78**	**97.78**

- ✓ **Nezha achieves high accuracy in *AS*@1 (90%), *AS*@3 (97%)**
- ✓ **Nezha outperforms all the baseline approaches**
- ✓ **Each data source contributes to the effectiveness of Nezha**

Experiment Result

- Root cause result at inner-service level
  - Resource Type or Code Block

Table 4: Comparison of baselines at inner-service level

Approach	OnlineBoutique			TrainTicket		
	AIS@1	AIS@3	AIS@5	AIS@1	AIS@3	AIS@5
SBLD	14.28	17.85	17.85	15.56	22.22	24.44
LogFaultFlagger	19.64	21.42	21.42	15.56	24.44	24.44
PDiagnose	35.71	53.57	71.42	8.89	13.33	15.56
Nezha w/o $\mathcal{M}$	26.78	33.92	35.71	55.56	62.22	68.89
Nezha w/o $\mathcal{L}$	64.28	64.28	64.28	42.22	44.44	44.44
Nezha	**92.86**	**96.43**	**96.43**	**86.67**	**97.78**	**97.78**

- ✓ **Nezha achieves high accuracy in *AIS@1* (87%), *AS@3* (97%)**
- ✓ **Nezha outperforms all the baseline approaches**
- ✓ **Each data source contributes to the effectiveness of Nezha**

Experiment Result

- Efficiency of Nezha

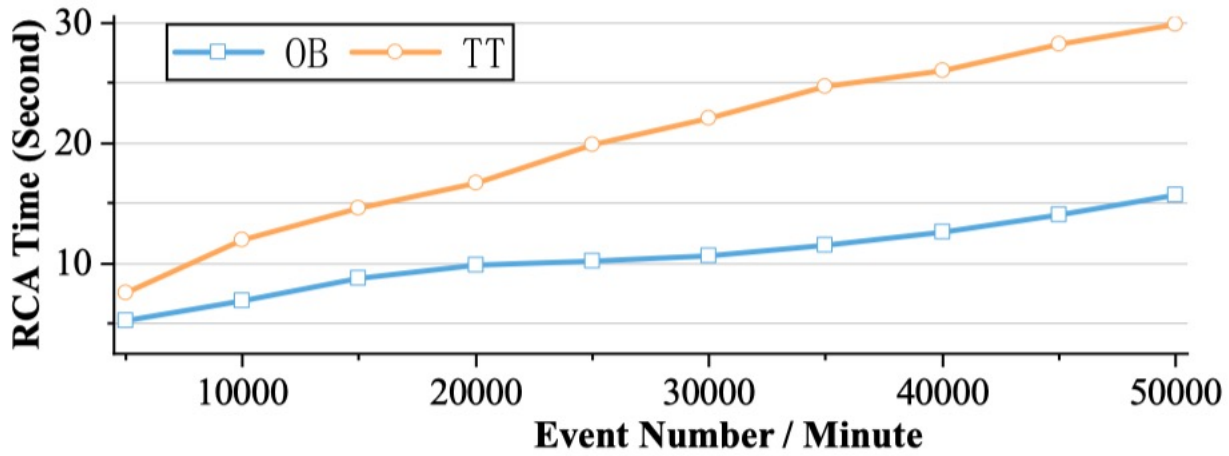
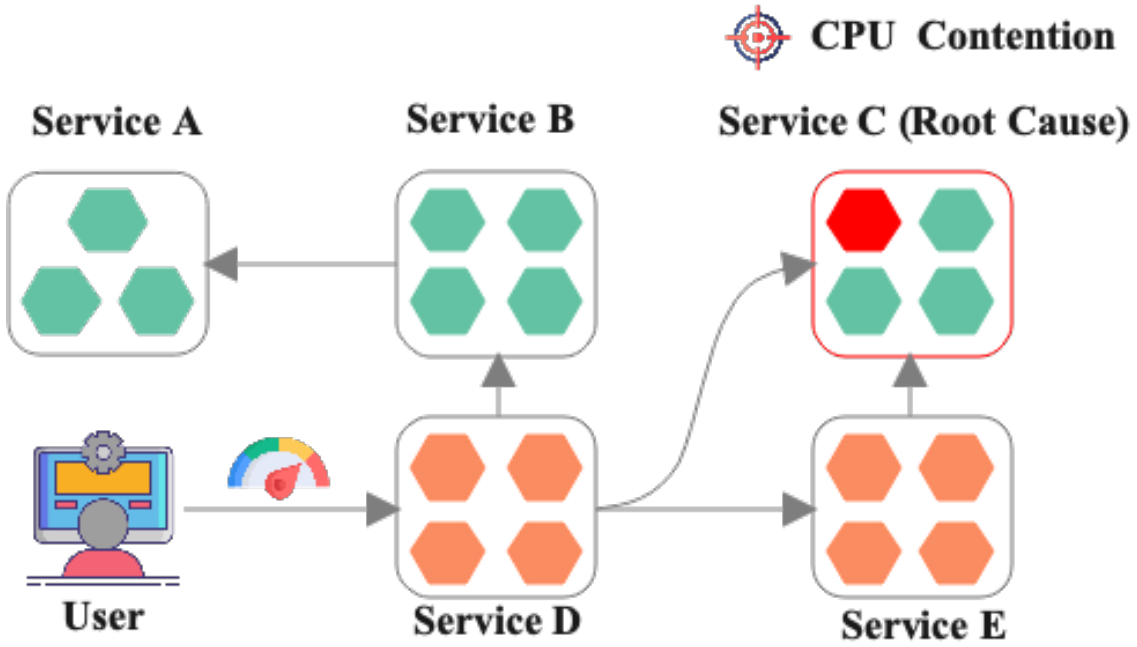


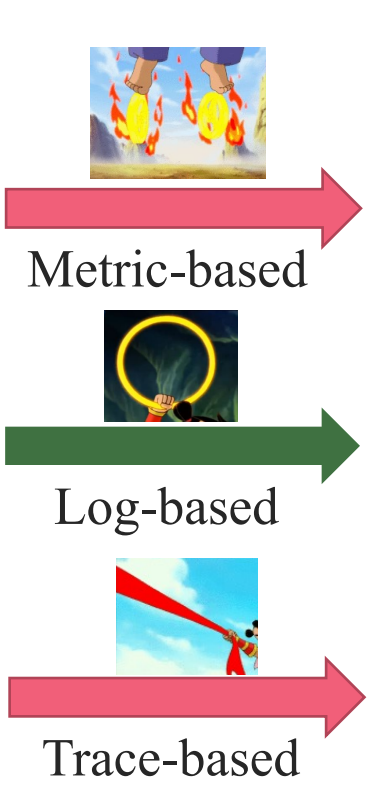
Figure 11: Change of diagnosis time with event number.

- ✓ **Diagnosis time of Nezha increases linearly with the number of events**
- ✓ **Nezha takes 16 seconds to determine root causes in a time window of 50,000 events**

Conclusion

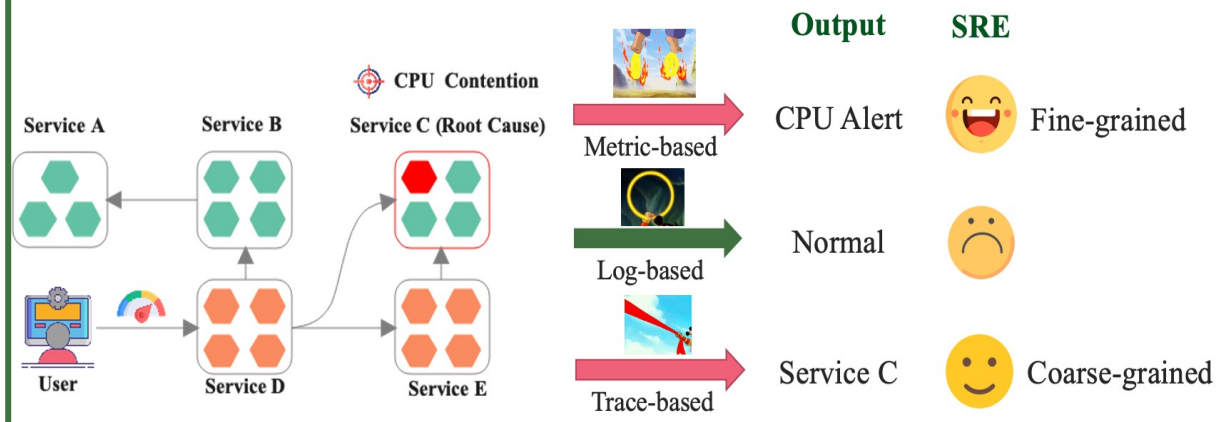
- Certain anomalies may not be apparent in some data sources
  - CPU contention is not obvious in logs





Conclusion

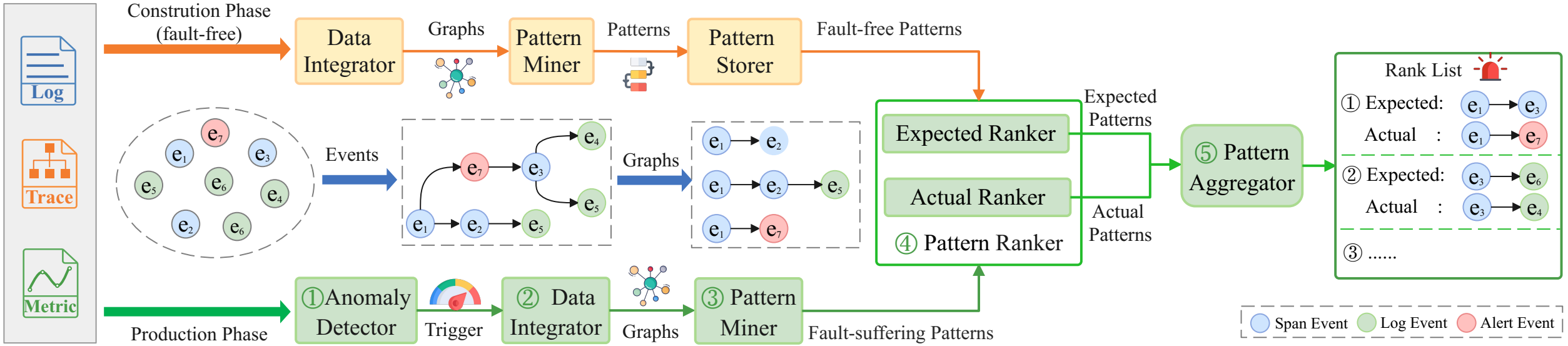
- Certain anomalies may not be apparent in some data sources
  - CPU contention is not obvious in logs



Conclusion

- Nezha: an unsupervised fine-grained RCA approach by incorporative analysis of multi-modal data in an interpretable manner

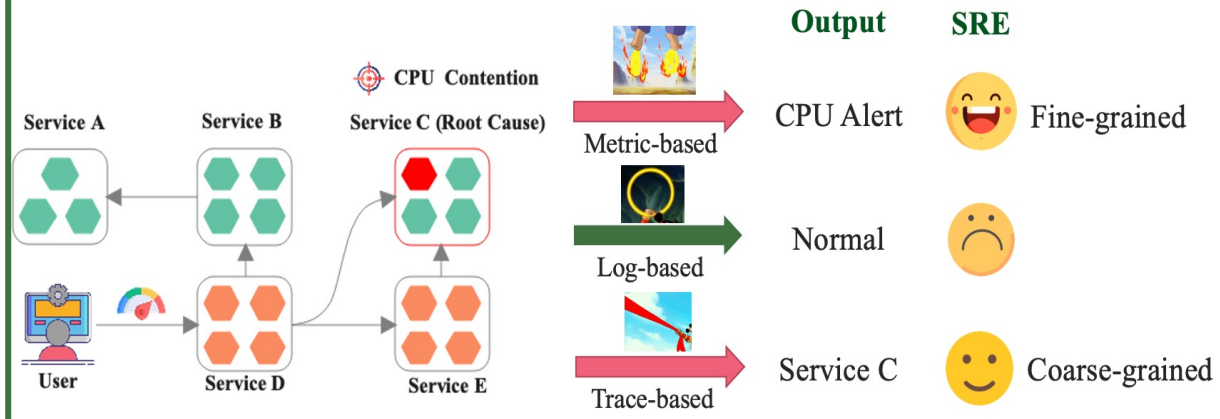
Fault-free Construction Phase



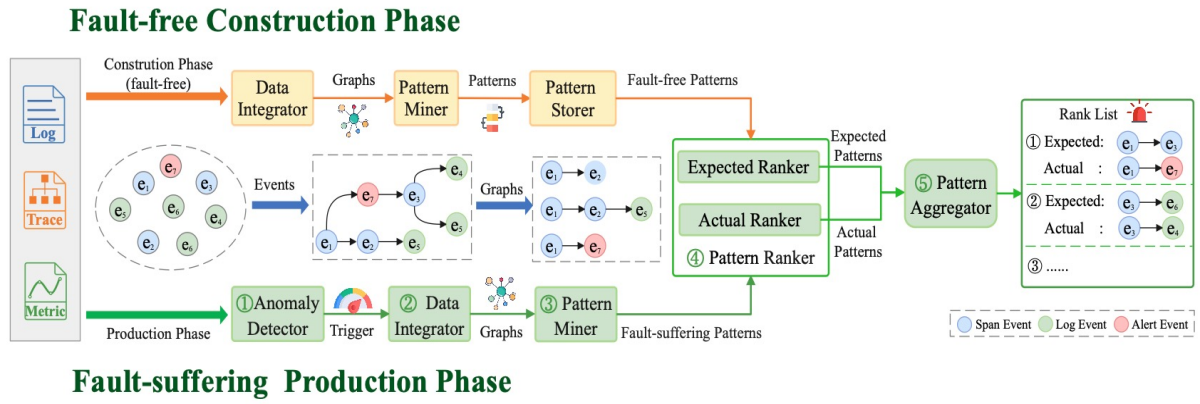
Fault-suffering Production Phase

Conclusion

- Certain anomalies may not be apparent in some data sources
  - CPU contention is not obvious in logs



- Nezha: an unsupervised fine-grained RCA approach by incorporative analysis of multi-modal data in an interpretable manner



Conclusion

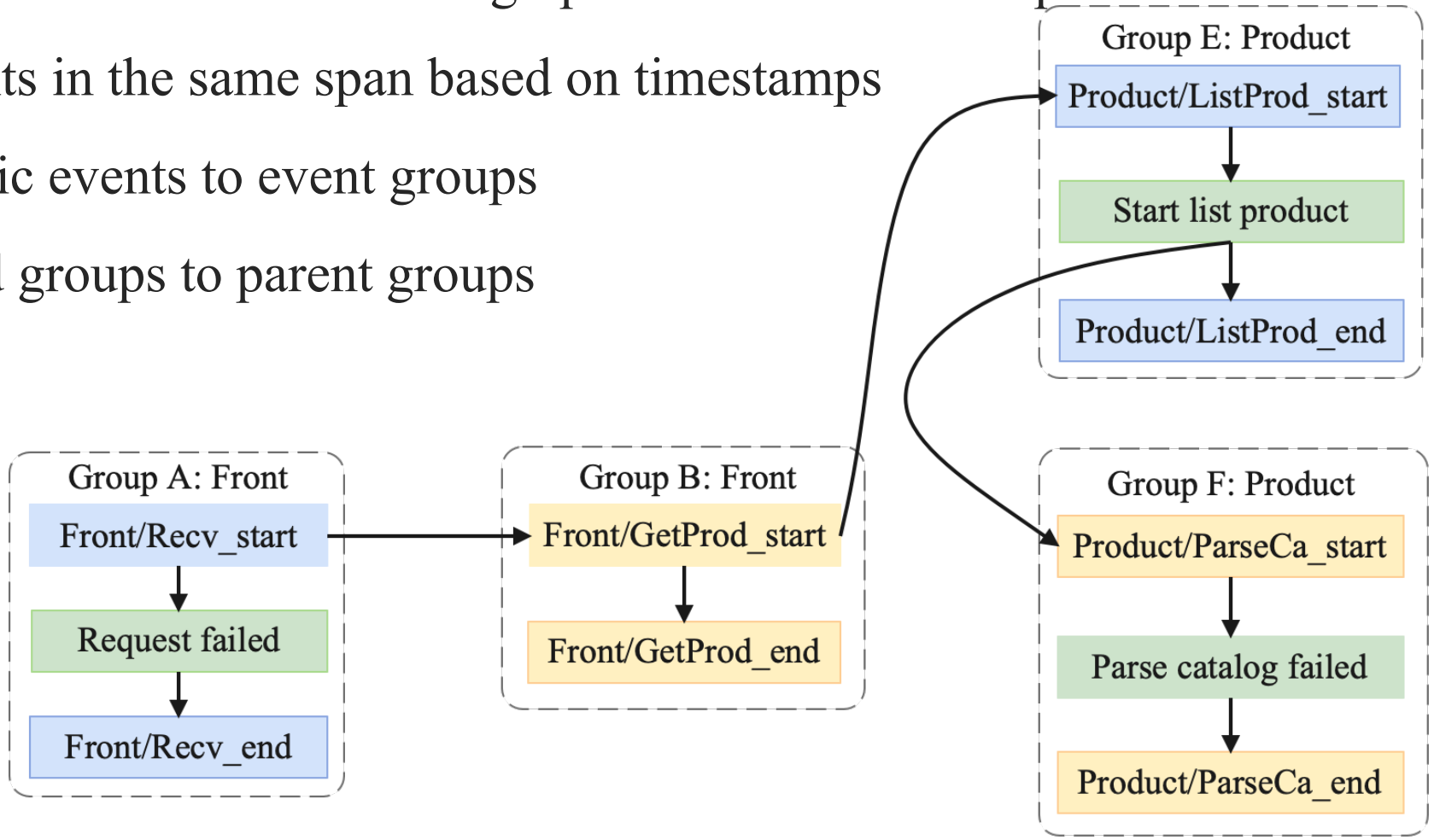
- Nezha: an unsupervised fine-grained RCA approach with multi-modal data

  ② Integrate multi-modal data into event graph based on timestamps and trace ID

  IV. Order events in the same span based on timestamps

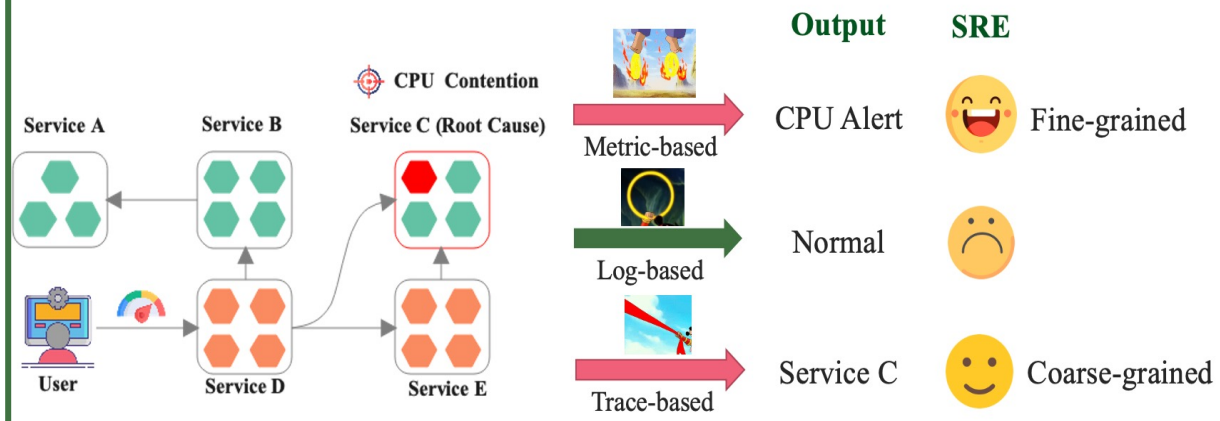
  V. Insert metric events to event groups

  VI. Insert child groups to parent groups

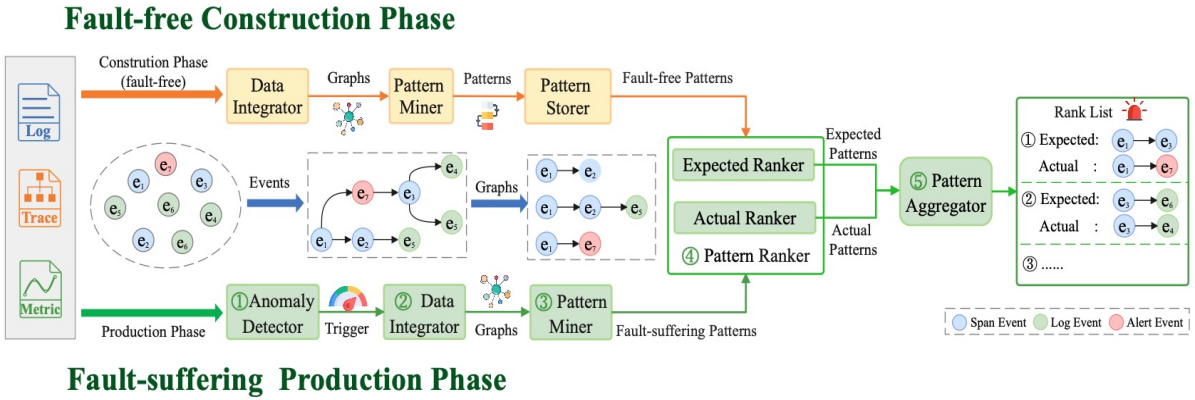


Conclusion

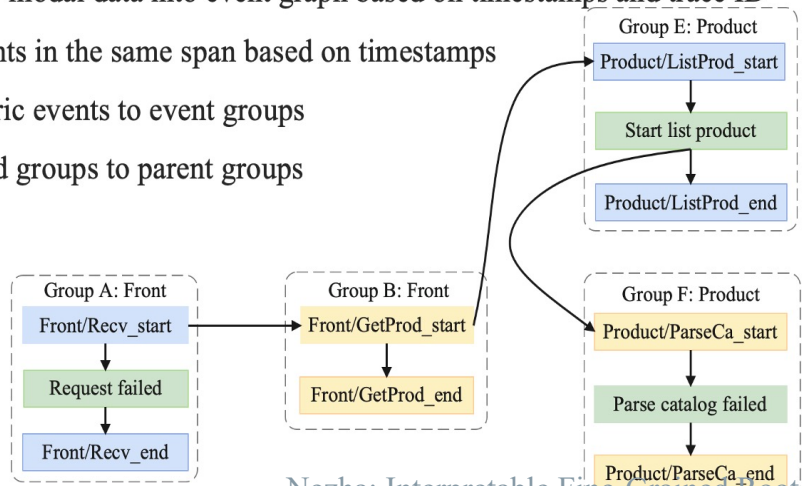
- Certain anomalies may not be apparent in some data sources
  - CPU contention is not obvious in logs



- Nezha: an unsupervised fine-grained RCA approach by incorporative analysis of multi-modal data in an interpretable manner



- Nezha: an unsupervised fine-grained RCA approach with multi-modal data
  - ② Integrate multi-modal data into event graph based on timestamps and trace ID
    - IV. Order events in the same span based on timestamps
    - V. Insert metric events to event groups
    - VI. Insert child groups to parent groups



Conclusion

- Root cause result at service level

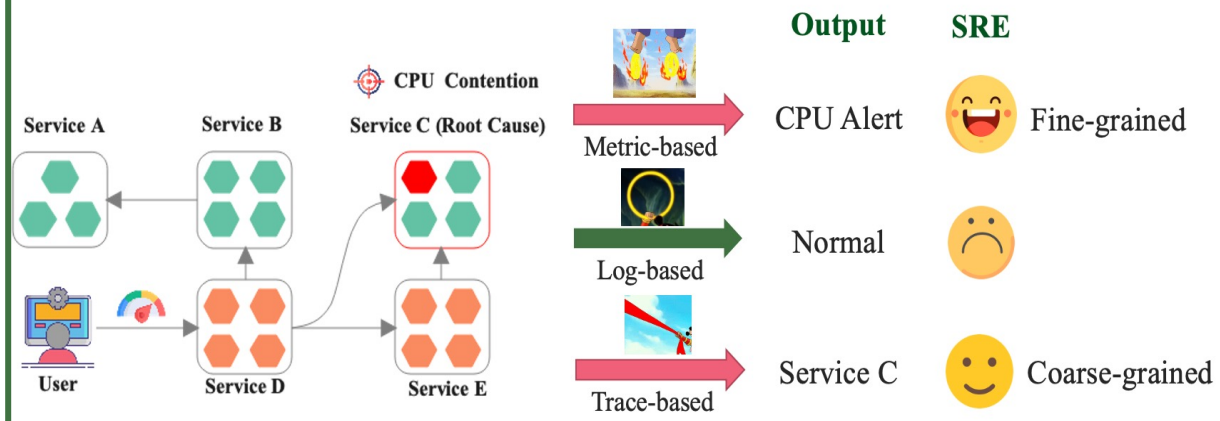
Table 3: Comparison of baselines at service level.

Approach	OnlineBoutique			TrainTicket		
	AS@1	*AS*@3	*AS*@5	*AS*@1	*AS*@3	*AS*@5
MicroScope	12.5	41.07	55.35	17.78	26.67	35.56
MicroRCA	16.07	62.5	92.75	20.00	31.11	44.44
SBLD	19.64	23.21	25.00	15.56	22.22	24.44
LogFaultFlagger	19.64	21.42	23.21	17.78	24.44	24.44
MicroRank	41.07	48.21	62.5	15.56	24.44	35.56
TraceAnomaly	30.35	33.92	48.21	13.33	28.89	33.33
PDiagnose	41.07	73.21	82.14	8.89	13.33	22.22
Nezha w/o $\mathcal{ML}$	14.28	17.85	17.85	6.67	8.89	11.11
Nezha w/o $\mathcal{M}$	26.78	33.92	35.71	55.56	62.22	68.89
Nezha w/o $\mathcal{L}$	64.28	64.28	64.28	42.22	44.44	44.44
Nezha	**92.86**	**96.43**	**96.43**	**86.67**	**97.78**	**97.78**

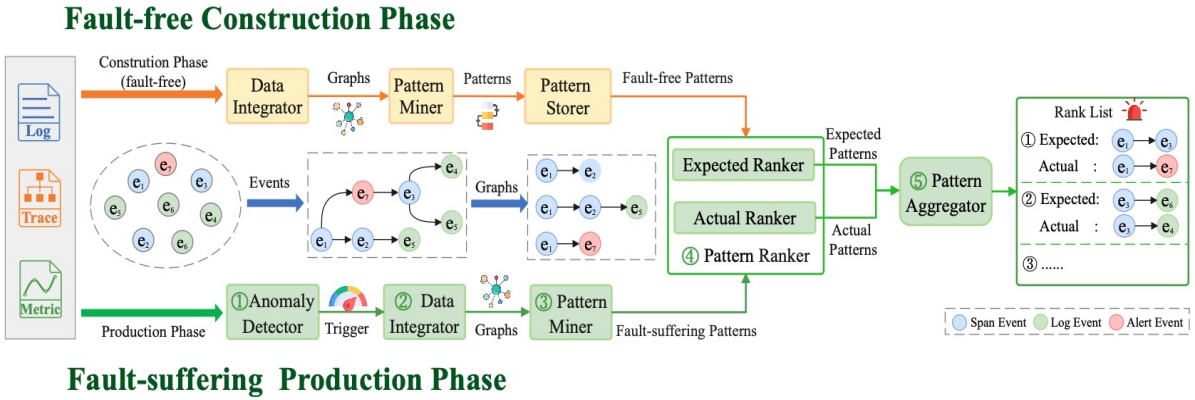
- ✓ **Nezha achieves high accuracy in *AS*@1 (90%), *AS*@3 (97%)**
- ✓ **Nezha outperforms all the baseline approaches**
- ✓ **Each data source contributes to the effectiveness of Nezha**

Conclusion

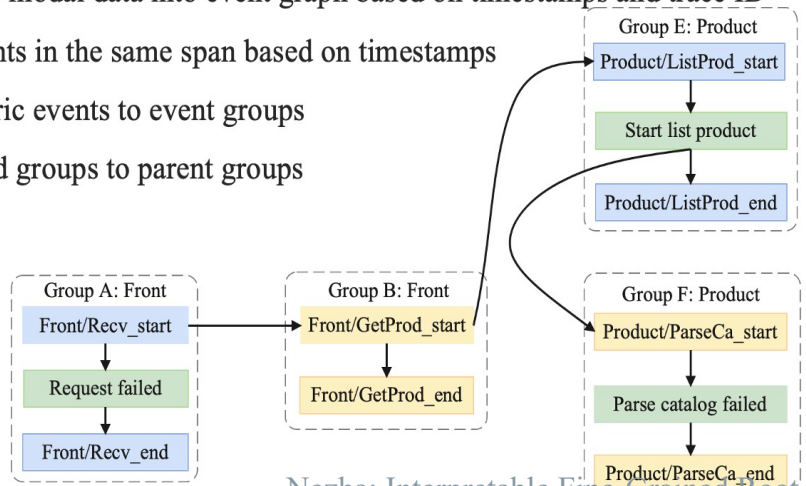
- Certain anomalies may not be apparent in some data sources
  - CPU contention is not obvious in logs



- Nezha: an unsupervised fine-grained RCA approach by incorporative analysis of multi-modal data in an interpretable manner



- Nezha: an unsupervised fine-grained RCA approach with multi-modal data
  - ② Integrate multi-modal data into event graph based on timestamps and trace ID
    - IV. Order events in the same span based on timestamps
    - V. Insert metric events to event groups
    - VI. Insert child groups to parent groups



- Root cause result at service level

Table 3: Comparison of baselines at service level.

Approach	OnlineBoutique			TrainTicket		
	AS@1	*AS@3*	*AS@5*	*AS@1*	*AS@3*	*AS@5*
MicroScope	12.5	41.07	55.35	17.78	26.67	35.56
MicroRCA	16.07	62.5	92.75	20.00	31.11	44.44
SBLD	19.64	23.21	25.00	15.56	22.22	24.44
LogFaultFlagger	19.64	21.42	23.21	17.78	24.44	24.44
MicroRank	41.07	48.21	62.5	15.56	24.44	35.56
TraceAnomaly	30.35	33.92	48.21	13.33	28.89	33.33
PDiagnose	41.07	73.21	82.14	8.89	13.33	22.22
Nezha w/o $\mathcal{ML}$	14.28	17.85	17.85	6.67	8.89	11.11
Nezha w/o $\mathcal{M}$	26.78	33.92	35.71	55.56	62.22	68.89
Nezha w/o $\mathcal{L}$	64.28	64.28	64.28	42.22	44.44	44.44
Nezha	**92.86**	**96.43**	**96.43**	**86.67**	**97.78**	**97.78**

- ✓ Nezha achieves high accuracy in *AS@1* (90%), *AS@3* (97%)
- ✓ Nezha outperforms all the baseline approaches
- ✓ Each data source contributes to the effectiveness of Nezha

中山大學
SUN YAT-SEN UNIVERSITY

Thank you !

Q & A

I am looking for post-doc opportunities

https://yuxiaoba.github.io
yugb5@mail2.sysu.edu.cn

https://github.com/IntelligentDDS/Nezha