FluxInfer: Automatic Diagnosis of Performance Anomaly for Online Database System

Ping Liu¹, Shenglin Zhang², Yongqian Sun², Yuan Meng¹, Jiahai Yang^{1,3}, Dan Pei¹







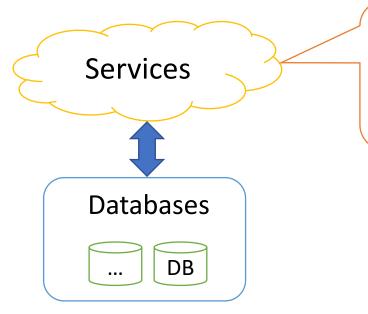


Background Design Evaluation Case Study



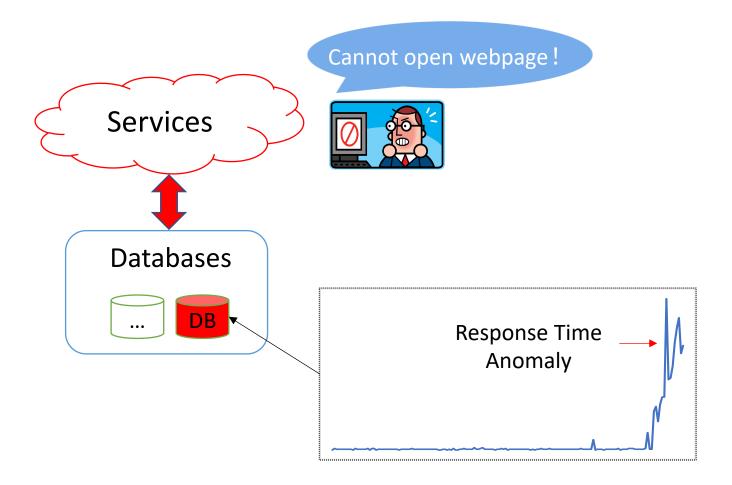
Background Design Evaluation Case Study

Database anomaly



The services need databases to support their mission-critical and real-time applications

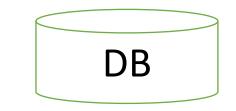
Database anomaly



Database anomaly

How to diagnose the **anomalies**?

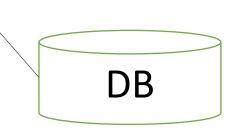
• To rapidly diagnose anomalies and trigger mitigation, database operators monitor hundreds of KPIs (Key Performance Indicator) of the database.



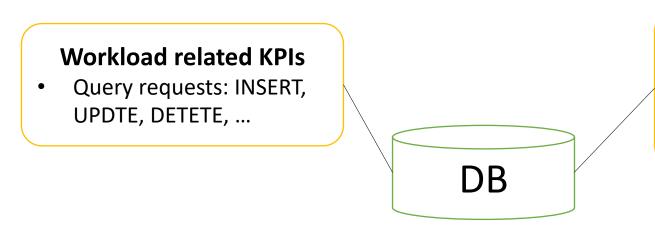
• To rapidly diagnose anomalies and trigger mitigation, database operators monitor hundreds of KPIs (Key Performance Indicator) of the database.

Workload related KPIs

• Query requests: INSERT, UPDTE, DETETE, ...



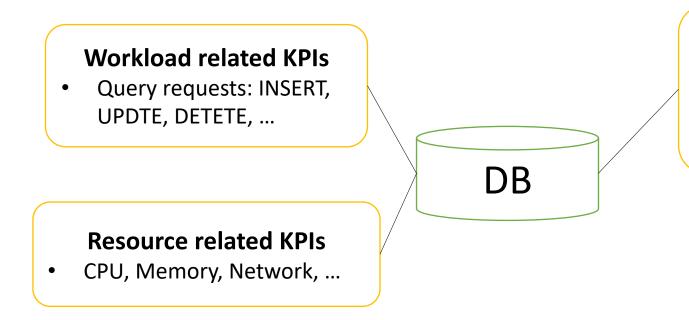
• To rapidly diagnose anomalies and trigger mitigation, database operators monitor hundreds of KPIs (Key Performance Indicator) of the database.



Storage-Engine-IO related KPIs

- Physical IO: The actual number of physical bytes involved in IO related operations.
- Logic IO: The actual number of virtual pages involved in IO related operations.

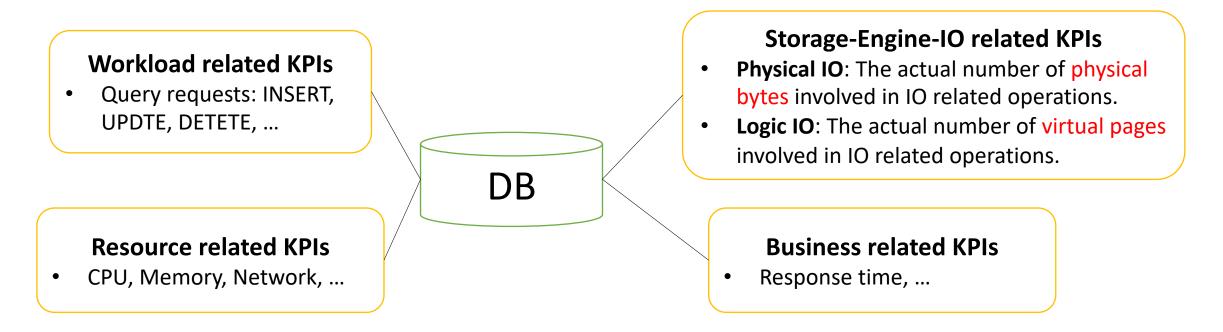
• To rapidly diagnose anomalies and trigger mitigation, database operators monitor hundreds of KPIs (Key Performance Indicator) of the database.



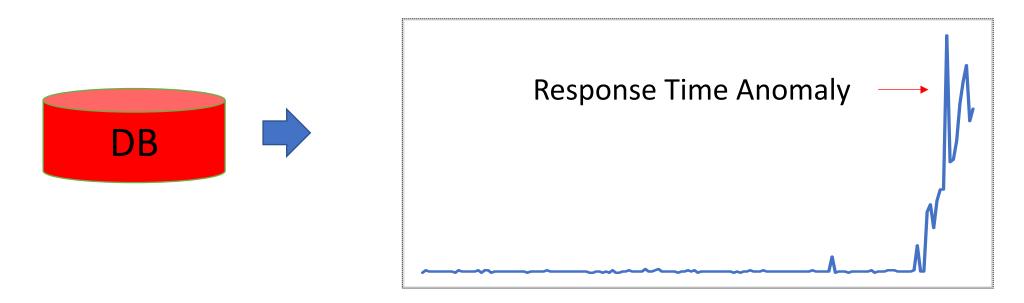
Storage-Engine-IO related KPIs

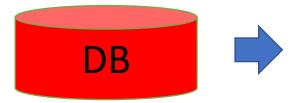
- Physical IO: The actual number of physical bytes involved in IO related operations.
- Logic IO: The actual number of virtual pages involved in IO related operations.

• To rapidly diagnose anomalies and trigger mitigation, database operators monitor hundreds of KPIs (Key Performance Indicator) of the database.

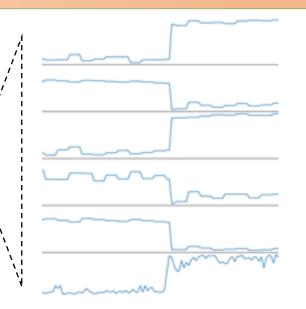


• When a database anomaly arises (response times are too slowly), some KPIs manifest anomalous patterns.





~~ A	1		
	<u> </u>		
		I	
		<u></u>	
	i .		
_ 60 _ (\sim	~~
		1	
	i		
		huma	
		1 m	~~~
	i	pa =	č
		1 m	n~~
	· · · · ·	<i>Jc</i> ² ·	-
multi many many many many many many many many		\	
	-	han	~~
	1	$\mu \sim$	• 0.0
	1	M.	M~~
Munder and the second s	• ~ ~ ~	\	
a between the second as a second s	i	h	~~
Mun My Marka Mar	www	m	um.
Mar May and a star from the source and the second start of the sec		0.144	
Palan Malanda Mark Mark Mark Mark Mark Mark Mark Mar	ww	mm	
And the second sec			
	-	han	~
		١.	
A M M M M M M M M M M M M M M M M M M M		Vm	~
	www.	mm	m
		han	
•	1		
Manufactures and		m	~
		han	
Management and a start and a			~~~~
and the second and the second se	h		
Man May Mark Mark Mark Mark Mark Mark Mark Mark	www.	10- A-	1.00
		V	
v		han	
Market	m	、 、	
	i —	han	~~
man man hand hand the man man hand have a second hard the seco			
March			
· · · · · · · · · · · · · · · · · · ·	1		
hundringer		mm	·
a manufacture and a second	·	1	
month and the second se	i .		
had a	······	m	~~~
	mm		
	1	min	~~~
	:		
Mr. Marine Ma	mm	1-	
	mm	him	i de la compañía de
Multimeter manufacture and	h	Vinne ///////////////////////////////////	****
	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		<del>~~~</del>
	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		~~~
man man man and a second a second and a second			
Manager and the second se		hanne James Lanne	
Manager and the second se		hand hand hand hand	



Anomalous patterns

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~		
	~~~	
	Murringe	
	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	
	h	
	from	
Many Market M	~~~~~	
	12mm	
Man Man Mar	J	
· · · · · · · · · · · · · · · · · · ·	man	
Mun Malage March March & Baren March March & Baren March & Baren & Bar	Month	/
Murthapparant and Marchapparant and and the second	WWW has made	
Market Marke	many many	
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	many	
A of month many as me was a second as a	h	
Mun May and Mark Mark Mark Mark Mark Mark Mark Mark	Mar Martin	
	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	
	han	
Menter and the second s	man	
manna and a second a s	. ~	
Mu Jugaland Mark Mark of Mark and	nMymm.	
	man	
	h	
· · · · · · · · · · · · · · · · · · ·	han	
marine have been a second and the second	h-h-	
and the second	~~~~~	
handreamantheantheantheantheantheantheantheanthe	m	
month and a second for and the second	Ann and	
hardware Alleria and a start a	mum	
Muy man	unit in the second seco	
with more and the second secon	- "Mulmon	
and a second state a protocol definition of the second state of th	Marrie Ma	
mannantermannanter	mont	
he dear and a second se	many	
	many	
my here was a second w	miny	
	Land	

#### **Root cause related KPI**

• These KPIs indicate the root cause of the database anomaly and thus is closely related to the root cause

#### Symptom KPI

• These KPIs are not directly related to the root cause

#### **Root cause related KPI**

• These KPIs indicate the root cause of the database anomaly and thus is closely related to the root cause

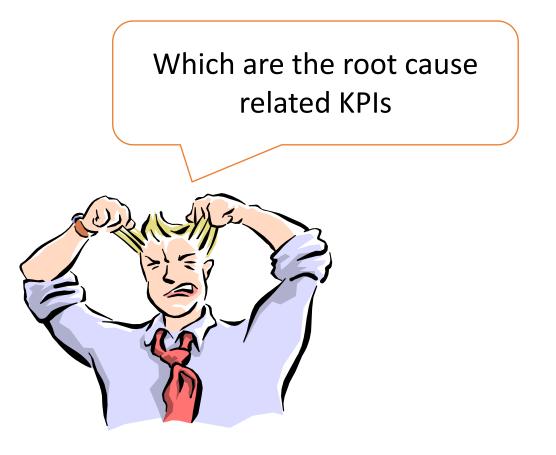
#### Symptom KPI

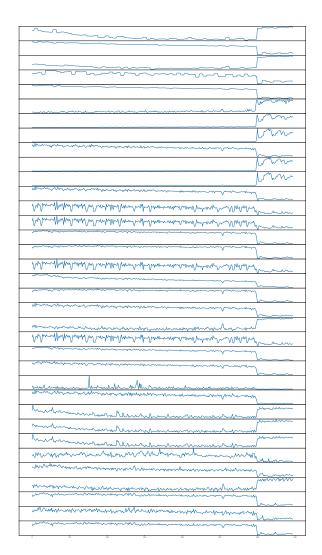
• These KPIs are not directly related to the root cause

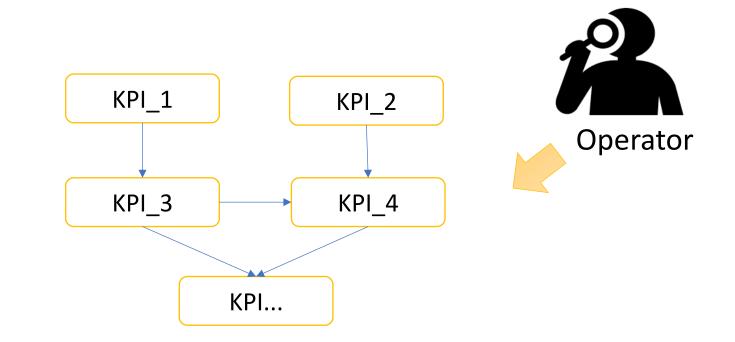
#### Example

- A sudden increase in workload leads to a database anomaly.
- The QPS KPI, which represents the workload of the database, is a **root cause related KPI**.
- On the other hand, the increase in workload also leads to a sudden increase in CPU utilization, which is represented by the CPU_USAGE KPI. Consequently, CPU_USAGE KPI is a symptom KPI.

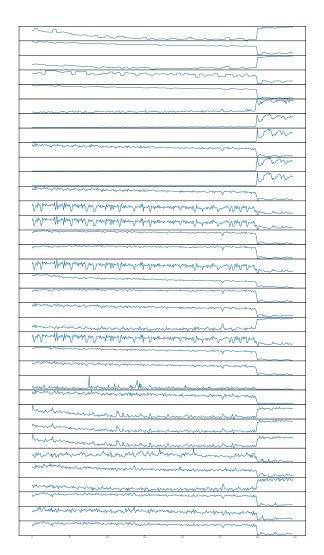
	1
	<u></u>
	J
	<u> </u>
	1_
	Murring
	hum
	mm
hand the second s	) <u> </u>
	ham
·····	
	from
	han
Man May and a start way a second way and a second	mm
have been and and a second a second a second a	ha a co
made a second and a second and a second	1
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	h
of month other ments as mentanes been all a sure an an a set of the second	h
a for the second and	mmm
	L
	han
here we have the second s	
man and the second s	
Man May Man Mark Mark Mark Mark Mark	
	mm
	h
	han
Martin Ma	
www.www.www.www.www.www.www.www.	\
hundren of her and	min
madam.	m
had as	m
and the second s	/
www.man.when.when.when.when.	Muham
Marken was a second was	Lumm
man man man man man man and the second	Junio
Martin	1
Make market and the second and the second se	n
Martin	1 Marine

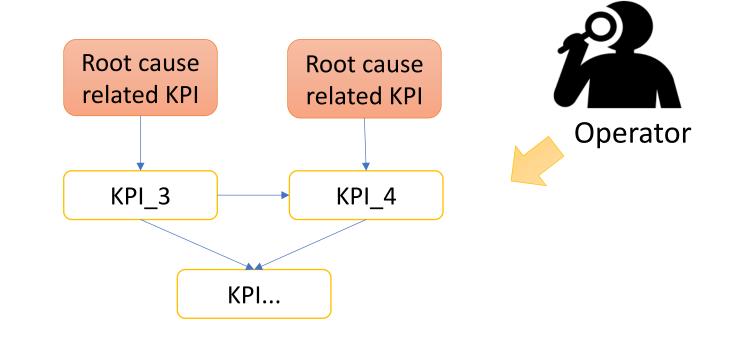






Operators **manually** infer the dependency relationships of these KPIs, which is highly dependent on the domain knowledge and experience





Then localize the root cause related KPIs based on the dependency relationships

However, it is **time-consuming** to manually diagnose a database anomaly even for a **highly experienced** operator, causing the services to suffer from performance degradation for a long period.

	hanne
harding a	- Andrew -
man man have have have a second and have a secon	- And and a second
Martin Ma	mm
moundation	mmm
Mohammen Management	man parman
man man was and man was a second	Munum
Marine Ma	mining
Make man and a second and a sec	mm Mumm
Martin Ma	mung

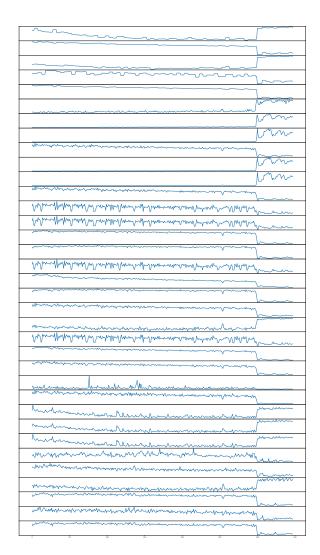
dependency relationships

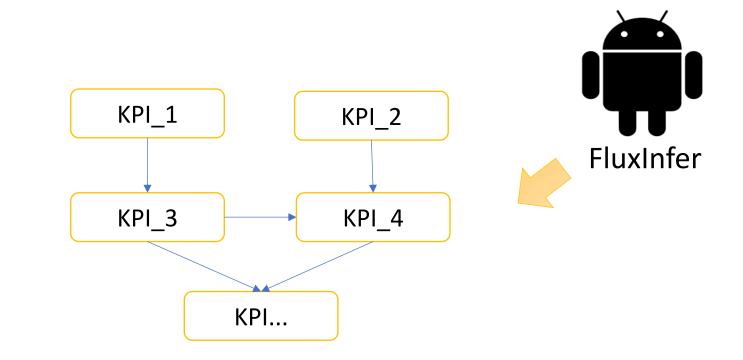
Core idea

	1~~~~
	- 1_
	7
	Munit
	J.
Manufacture and the second second	N www.
	han
	from
	from
Murray Marray Marray Marray Marray	η
Mar Makalan Mar	The days
Man Makaka Marka Mar	M.
	7 minun
m was a second was a second was a second a secon	han
a de mande de mander de las services des la de la d	han
Mu May and Mark Mark Mark Mark Mark	man
	2
	han
Were Marine Ma	han
monorman management	
Munday and a share all have a shore a shore where the share the sh	The a car
	7
Many Market	2
	han
Martin	M
handressee	Manana
mon man and the second second	-
much man and and and and and and and and and a	~
harden har	~~~~~
more many many many many	Malan
May market and the second and the second sec	7-
many many many many many many many many	
man and the second and the second sec	m M
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	h
	<u>"Ummm</u>
	Imm



#### Core idea

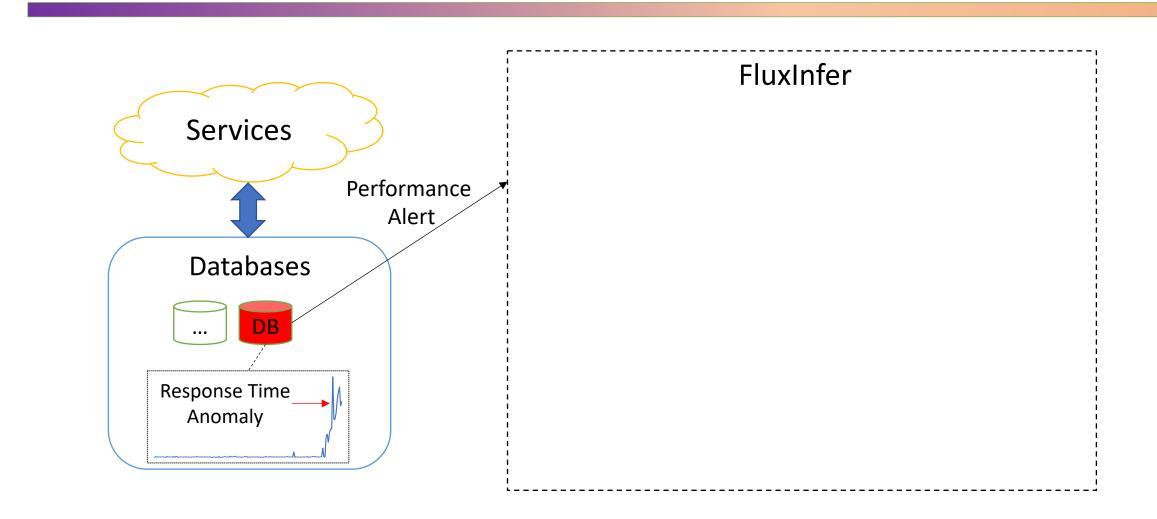


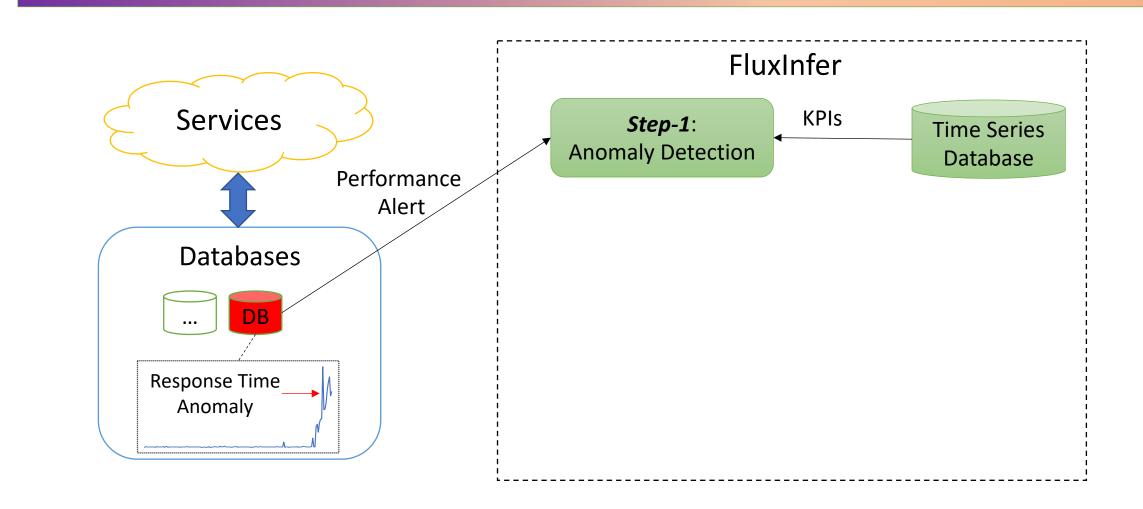


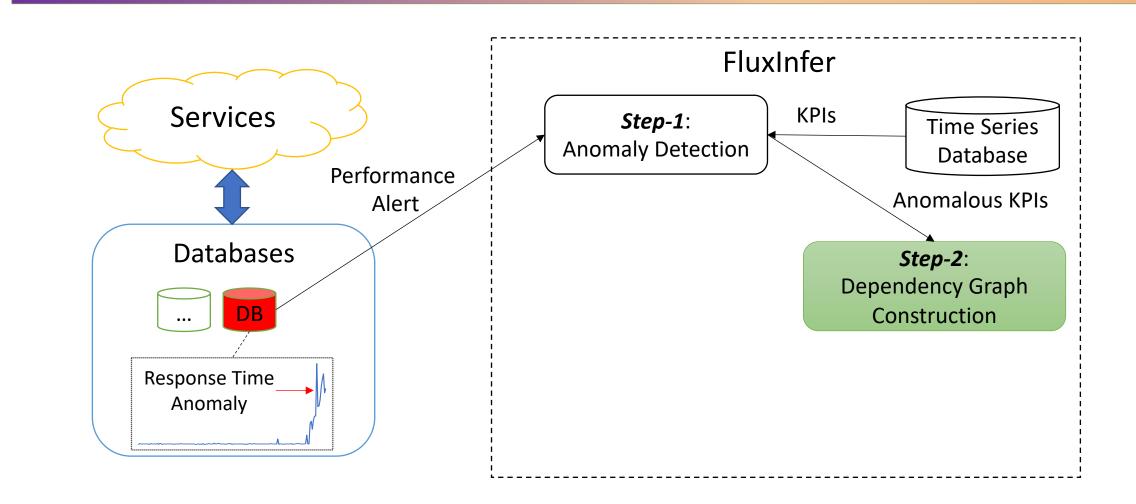
FluxInfer can automatically construct the dependency relationships of anomalous KPIs accurately and localize root cause related KPIs



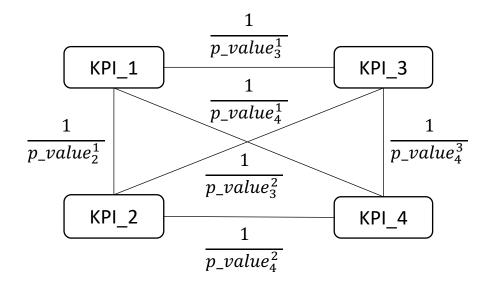
Background Design Evaluation Case Study





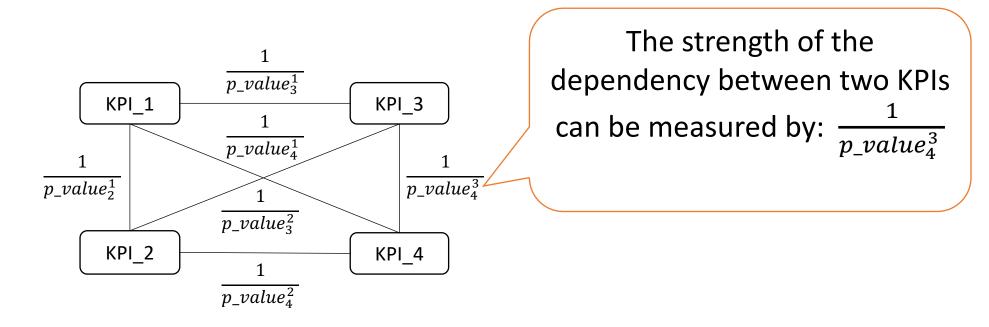


#### Dependency Graph

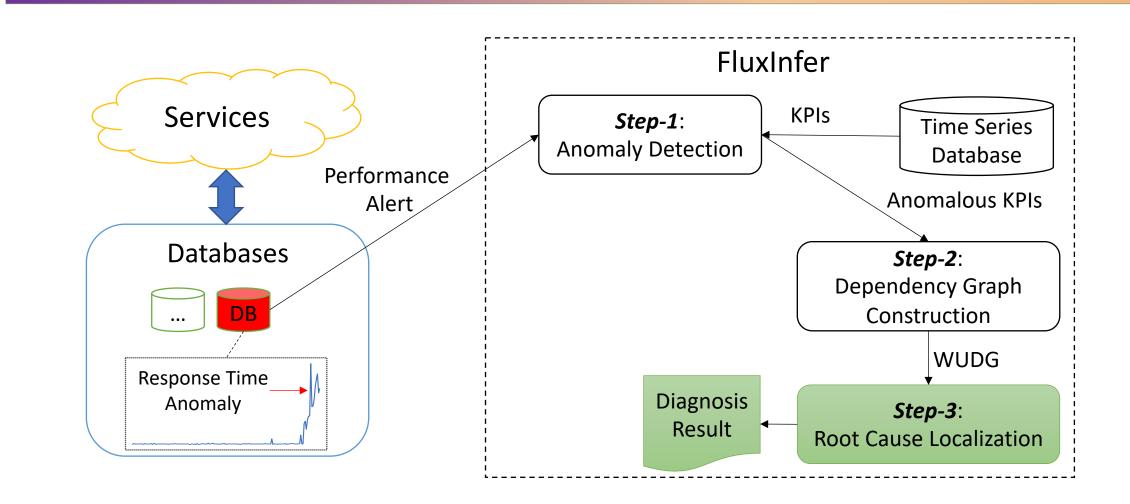


Weighted undirected dependency graph (WUDG)

#### Dependency Graph

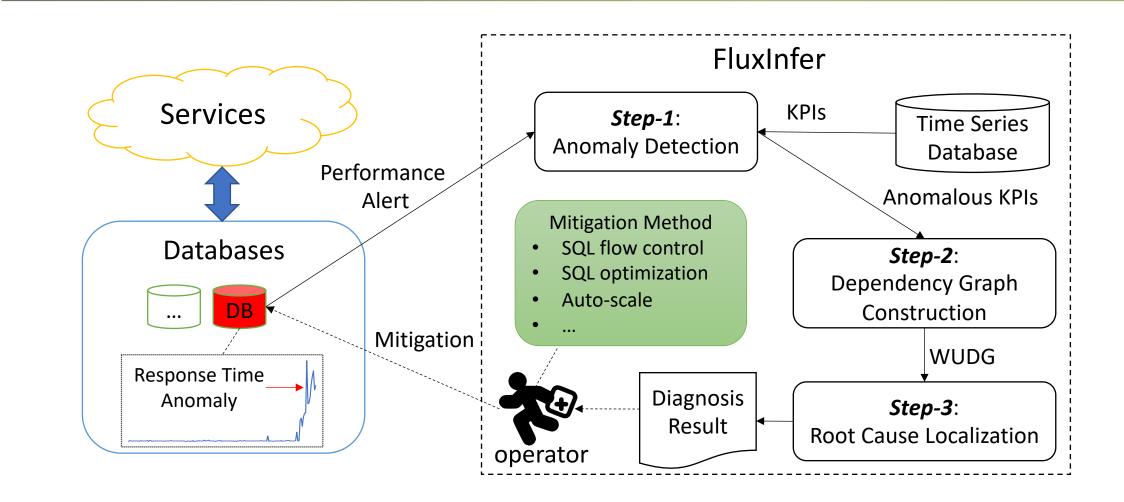


Weighted undirected dependency graph (WUDG)



#### **Root Cause Localization**

- Due to that the root cause of a database system can quickly spread and lead to more and more anomalous KPIs, the root cause has the largest influence on the dependency graph among KPIs.
- Therefore, we suppose that the root cause related KPI is the KPI who has the largest influence on the dependency graph.
- The weighted PageRank algorithm are used to measure the influences of nodes in a weighted undirected graph, the possible root cause related KPIs are the KPIs ranked at the top.





#### Background Design Evaluation Case Study

#### Dataset

- We constructed a testbed to generate accurately labeled anomalies of database performance for evaluation.
- We injected five different types of anomalies: CPU saturation, network congestion, IO saturation, memory saturation, anomalous workload.

Type of anomaly	Number	Description				
CPU Saturation	30	Invoke stress-ng, which starts N workers that perform various matrix operations on floating point values.				
Network Congestion 30		Simulate network congestion by adding an artificial 500-milliseconds delay to every traffic over the				
Network Congestion	50	network via Linux's tc (Traffic Control) command.				
IO Saturation 30		Invoke stress-ng, which starts N workers that perform a mix of sequential, random and memory mapped				
10 Saturation	30	read/write operations as well as forced sync'ing and cache dropping.				
Memory Saturation	30	Invoke stress-ng, which starts N workers that grow their heaps by reallocating memory.				
Anomalous workload 30		Greatly increase the rate of transactions and the number of clients simulated by OLTPBenchmark				
Anomaious workioau		(150 additional terminals with transaction rate of 50,000).				

TABLE II: Details of injected 150 database anomalies for evaluation.

#### **Evaluation metric**

- AC@k represents the accuracy that top K results include the root causes related KPIs for all anomaly cases.
- Avg@k represents the overall performance.

$$Avg@k = \frac{1}{k} \sum_{1 \le j \le k} AC@j.$$

Algorithm	Relationships Learning	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

#### TABLE III: The evaluation results of different algorithms

Algorithm	Relationships Learning	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

#### TABLE III: The evaluation results of different algorithms

Algorithm	Relationships Learning	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

TABLE III: The evaluation results of different algorithms

In FluxInfer-with-CUSUM, we replaced the robust anomaly detection algorithm of FluxInfer with the CUSUM algorithm

Algorithm	Relationships Learning	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

#### TABLE III: The evaluation results of different algorithms

- In FluxInfer-without-AD, we removed the robust anomaly detection algorithm of FluxInfer.
- All KPIs (normal KPIs and anomalous KPIs) are used to construct the dependency graph and localize the root cause.

Algorithm	<b>Relationships Learning</b>	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

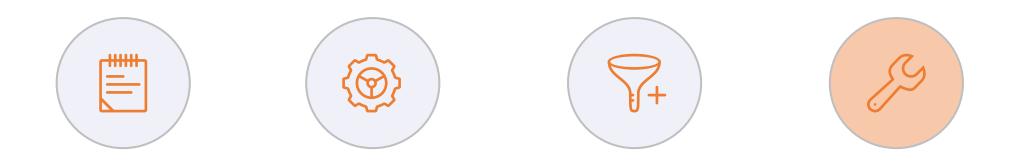
#### TABLE III: The evaluation results of different algorithms

• The results of FluxInfer-with-CUSUM and FluxInfer-without-AD demonstrate the effectiveness of our robust anomaly detection design.

Algorithm	<b>Relationships Learning</b>	Root Cause Inference	AC@1	AC@2	AC@3	AC@5	Avg@5
PAL [20]	N/A	Anomaly Time Order	0.09	0.12	0.14	0.20	0.14
CauseInfer [3]	PC Algorithm	Deep First Search	0.12	0.20	0.22	0.28	0.21
CloudRanger [4], MS-Rank [7]	PC Algorithm	Second-order Random Walk	0.08	0.19	0.27	0.36	0.24
Microscope [6]	PC Algorithm	Traversing+Pearson Correlation	0.06	0.11	0.16	0.24	0.15
MicroRCA [21]	PC Algorithm	Personalized PageRank	0.08	0.17	0.30	0.38	0.25
MonitorRank [18], TON18 [19]	PC Algorithm	Random Walk	0.08	0.16	0.28	0.39	0.24
MicroCause [5]	PCTS	TCORW	0.23	0.38	0.47	0.60	0.44
FluxInfer	WUDG	Weighted PageRank	0.43	0.69	0.90	0.95	0.77
FluxInfer-with-CUSUM	WUDG	Weighted PageRank	0.23	0.40	0.62	0.73	0.53
FluxInfer-without-AD	WUDG	Weighted PageRank	0.13	0.20	0.32	0.50	0.38

#### TABLE III: The evaluation results of different algorithms

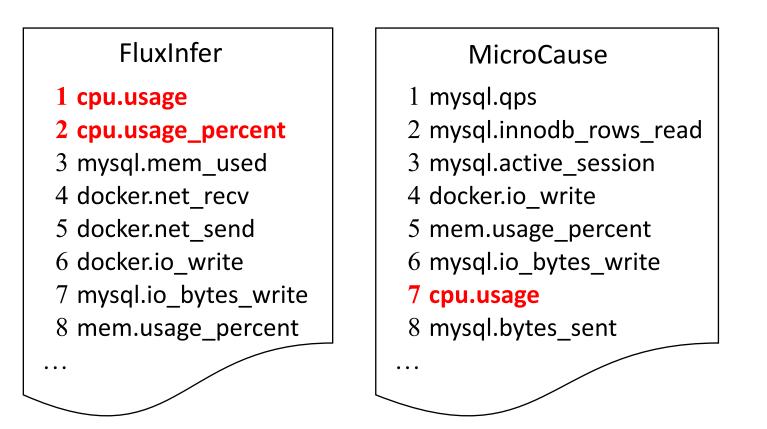
• Further, we can see that the performance of FluxInfer-with-CUSUM is better than that of FluxInfer-without-AD, which demonstrates that anomaly detection is necessary for the design of FluxInfer.



#### Background Design Evaluation Case Study

#### Case study

• The diagnosis results of FluxInfer and MicroCause baseline for a CPU saturation case. The bold red font represents the root cause related KPI.



### Case study

- The diagnosis results of FluxInfer and MicroCause baseline for a CPU saturation case. The bold red font represents the root cause related KPI.
  - The diagnosis result of FluxInfer shows that the root cause related KPIs are ranked top 1 and top 2, respectively.
  - However, the root cause related KPI is ranked top 7 in the diagnosis result of MicroCause baseline.
  - The other eight baselines even cannot rank the root cause related KPIs in top 10.

### Conclusion

- This paper presents FluxInfer, which can automatically and accurately localize the root cause related KPIs of online database performance anomalies.
- We propose an algorithm to automatically construct a Weighted Undirected Dependency Graph (WUDG) to accurately represent the dependency relationships of anomalous KPIs.
- We propose to use a weighted PageRank algorithm to traverse WUDG, which can accurately localize root cause related KPIs.
- Detailed evaluation experiments on our testbed show that the AC@3, AC@5, and Avg@5 of FluxInfer are 0.90, 0.95, and 0.77, outperforming nine baselines by 64%, 60%, and 53% on average, respectively.

# Thank you! Q&A



liuping15@mails.tsinghua.edu.cn

IPCCC 2020