Diagnosing Root Causes of Intermittent Slow Queries in Cloud Databases

Minghua Ma, Zheng Yin, Shenglin Zhang, Sheng Wang, Christopher Zheng, Xinhao Jiang, Hanwen Hu, Cheng Luo, Yilin Li, Nengjun Qiu, Feifei Li, Changcheng Chen, Dan Pei
Agenda

• **Keywords:** Cloud Databases/Slow Queries/Intermittent Slow Queries

• **Why?** Challenges & Motivations

• **What?** iSQUAD

• **How?** Database Autonomy Service
Cloud Databases

Cloud + Database = Cloud Compute + Database

Cloud: Resource Pool
Computer: Compute + Storage

Data: Produce -> Process -> Store -> Consume

OLTP Database
Slow queries result from internal reasons:

• nature of complexity
• lack of indexes
• poorly-written SQL statements...
Intermittent Slow Queries (iSQs)

SQL B Execution Time

E₁

E₂

0 1
Intermittent Slow Queries (iSQs)

Slow queries result from external reasons:
- Instance CPU intensive workload
- Host I/O bottleneck
- Accompanying slow SQL...
Motivations: More or Less

> 100 BUs  > 1,000,000 DBs

More
- Availability
- Security
- Stability
- Scalability
- Performance

Less
- Human Labor
- Machines
- ...

x1000

< 50 DBAs
Motivations: Definition of iSQs

\( X_t \): one particular query execution time

\[ X_t > 1 \land P(X_t > 1) < 0.01: \text{iSQs} \]

Thresholds are set empirically on Alibaba Database
Impact of iSQs

Most of iSQs are interactive queries

iSQs -> Poor user experience -> Revenue loss
Diagnosing Root Causes of iSQs in the Cloud

Multiple database instances may reside on the same physical machines, which can cause resource contentions.
Complexity infrastructures of cloud databases make it harder for DBAs to diagnose root causes of iSQs.
Outline

What's iSQ?

Why it's challenging?

How to diagnose it?

Evaluation
Challenges: Anomaly Diversity

Different types of database Key Performance Indicators (KPIs)

(a) Spike Up  (b) Spike Down  (c) Level Shift Up  (d) Level Shift Down

Current binary anomaly detectors generally overlook and over-generalize the types of anomalies.
Challenges: Labeling Overheads

Tens of thousands of iSQs per day in Alibaba Database

Scan hundreds of KPIs to find out the root cause of an iSQ

Manually labeling root causes is massive work; Reproducing known root causes in a testbed experiment is not feasible.
Challenges: Interpretable Models

Being able to explain or narrate what causes the problem when it arises is essential in cloud databases.

An inevitable trade-off exists between a model's accuracy and its interpretability to human.
iSQUAD

Intermittent Slow Queries
Anomaly Diagnoser
iSQUAD Overview

**Offline Analysis & Explanation**

Historical Intermittent Slow Queries & KPIs

- Anomaly Extraction
  - iSQ1
  - iSQ2
  - iSQ3
  - iSQ4
  - iSQ5
  - CPU
  - I/O

- Dependency Cleansing
  - iSQ1
  - iSQ2
  - iSQ3
  - iSQ4
  - iSQ5
  - CPU
  - I/O

- TOPIC
  - C1
  - iSQ1
  - iSQ2
  - iSQ3
  - iSQ4
  - iSQ5
  - CPU
  - I/O

- BCM Illustration
  - Case #1: iSQ1
  - In Cluster C1: iSQ1, iSQ2
  - Subspace:
    - Root Cause: DBA Labeling

**Online Root Cause Diagnosis & Update**

Incoming Intermittent Slow Queries & KPIs

- Anomaly Extraction
  - iSQx
  - CPU
  - I/O

- Dependency Cleansing
  - iSQx
  - CPU
  - I/O

- Cluster Assignment
  - iSQx is matched with C1
  - Case #1: iSQ1

- Root Cause

Optional Update

- Historical Intermittent Slow Queries & KPIs
  - CPU
  - I/O
  - iSQ1
  - iSQ2
  - iSQ3
  - iSQ4
  - iSQ5
iSQUAD Overview

Historical Intermittent Slow Queries & KPIs

Anomaly Extraction
- CPU
- I/O
- iSQ1
- iSQ2
- iSQ3
- iSQ4
- iSQ5

Dependency Cleansing
- CPU
- I/O
- iSQ1
- iSQ2
- iSQ3
- iSQ4
- iSQ5

TOPIC
- CPU
- I/O
- C1
- iSQ1
- iSQ2
- C2
- iSQ3
- iSQ4
- C3
- iSQ5

BCM Illustration
- Case #1: iSQ1
  - In Cluster C1: iSQ1, iSQ2
  - Subspace:
  - Root Cause: DBA Labeling

Incoming Intermittent Slow Queries & KPIs

Anomaly Extraction
- CPU
- I/O
- iSQx

Dependency Cleansing
- CPU
- I/O
- iSQx

Cluster Assignment
- iSQx is matched with C1
  - Case #1: iSQ1

Optional Update

Root Cause
iSQUAD Overview

Anomaly Extraction

CPU
iSQ1 ▲ ● ★
iSQ2 ▲ ● ★
iSQ3 ▲ ● ★
iSQ4 ▲ ● ★
iSQ5 ● ● ★

I/O

Dependency Cleansing

CPU
iSQ1 ▲ ○ ★
iSQ2 ▲ ○ ★
iSQ3 ▲ ○ ★
iSQ4 ▲ ○ ★
iSQ5 ○ ○ ★

I/O

TOPIC

CPU

I/O

Cluster Assignment

iSQx is matched with C1
Case #1: iSQ1

Root Cause: DBA Labeling

BCM Illustration

Case #1: iSQ1
In Cluster C1: iSQ1, iSQ2
Subspace:

Root Cause

Optional Update

Incoming Intermittent Slow Queries & KPIs

Historical Intermittent Slow Queries & KPIs
iSQUAD Overview

Anomaly Extraction
- CPU: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 
- I/O: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 

Dependency Cleansing
- CPU: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 
- I/O: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 

TOPIC
- CPU: 
  - C1: 
  - C2: 
  - C3: 
- I/O: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 

BCM Illustration
- Case #1: iSQ1
  - In Cluster C1: iSQ1, iSQ2
  - Subspace:
  - Root Cause: DBA Labeling

Optional Update

Incoming Intermittent Slow Queries & KPIs
- CPU: 
  - iSQx: 
- I/O: 
  - iSQx: 

Historical Intermittent Slow Queries & KPIs
- CPU: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 
- I/O: 
  - iSQ1: 
  - iSQ2: 
  - iSQ3: 
  - iSQ4: 
  - iSQ5: 

Cluster Assignment
- iSQx is matched with C1
  - Case #1: iSQ1

Root Cause
Anomaly Extraction

KPIs are important to locate iSQs' root causes. The anomaly types of KPIs should be paid attention to.
## Anomaly Extraction

### Multiple KPIs

- **Spike**
  - **Robust Threshold** [Sigmod 2018]
  - F1-Score: 98.7%
  - Time: 0.19s
  - dSPOT [KDD 2017]
  - F1-Score: 81.0%
  - Time: 15.11s

- **Level-Shift**
  - **T-Test**
  - F1-Score: 92.6%
  - Time: 0.23s
  - iSST [ISSRE 2018]
  - F1-Score: 60.7%
  - Time: 6.06s
One anomalous KPI is usually accompanied by another one or more anomalous KPIs.
Dependency Cleansing

Based on the association rule mining between two KPIs to determine whether the two KPIs have a correlation

$$\text{confidence}(A \rightarrow B) = \frac{|A \cap B|}{|A|}$$

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1-Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confidence</td>
<td>90.9</td>
<td>100</td>
<td>95.2</td>
</tr>
<tr>
<td>MI [Sigmod 2016]</td>
<td>100</td>
<td>40</td>
<td>57.1</td>
</tr>
<tr>
<td>Gain Ratio [Infocom 2016]</td>
<td>87.5</td>
<td>70</td>
<td>77.8</td>
</tr>
</tbody>
</table>
TOPIC: Type-Oriented Pattern Integration Clustering

Similar symptoms are correlated to the same root cause.
TOPIC: Type-Oriented Pattern Integration Clustering

- **KPI Type** – KPIs are classified into eight types by DBAs
- **Anomaly Pattern** – Similarity calculate by matching coefficient
- **Clustering main idea** – hierarchically merge similar pattern
Bayesian Case Model (BCM) is a framework for extracting prototypical cases and feature subspace [NeurIPS 2014].
BCM Illustration

- Initial labeling root cause
- Visualization case and feature (anomaly KPI) subspace
- New coming iSQ's root cause modification
- Labeling new clusters
iSQUAD Prototype Are Used in Alibaba Cloud Database

Offline Analysis & Explanation

Historical Intermittent Slow Queries & KPIs

Anomaly Extraction

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>iSQ1</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ2</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ3</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ4</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ5</td>
<td>★</td>
<td>★★</td>
</tr>
</tbody>
</table>

Dependency Cleansing

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>iSQ1</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ2</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ3</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ4</td>
<td>▲</td>
<td>★★</td>
</tr>
<tr>
<td>iSQ5</td>
<td>★</td>
<td>★★</td>
</tr>
</tbody>
</table>

TOPIC

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>iSQ1</td>
<td>★★</td>
</tr>
<tr>
<td>C2</td>
<td>iSQ2</td>
<td>★★</td>
</tr>
<tr>
<td>C3</td>
<td>iSQ3</td>
<td>★★</td>
</tr>
</tbody>
</table>

BCM Illustration

Case #1: iSQ1
In Cluster C1: iSQ1, iSQ2
Subspace:
Root Cause: DBA Labeling

Online Root Cause Diagnosis & Update

Incoming Intermittent Slow Queries & KPIs

Anomaly Extraction

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>iSQx</td>
<td>▲</td>
<td>★★</td>
</tr>
</tbody>
</table>

Dependency Cleansing

<table>
<thead>
<tr>
<th></th>
<th>CPU</th>
<th>I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>iSQx</td>
<td>▲</td>
<td>★★</td>
</tr>
</tbody>
</table>

Cluster Assignment

iSQux is matched with C1
⇒ Case #1: iSQ1

Root Cause

Optional Update
Outline

What's iSQ?

Why it's challenging?

How to diagnose it?

Evaluation
### iSQUAD Accuracy

#### iSQ ground truth labeled by DBAs

<table>
<thead>
<tr>
<th>No.</th>
<th>Root Cause</th>
<th>Offline</th>
<th>Online</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Instance CPU Intensive Workload</td>
<td>27.6%</td>
<td>34.5%</td>
</tr>
<tr>
<td>2</td>
<td>Host I/O Bottleneck</td>
<td>17.2%</td>
<td>17.2%</td>
</tr>
<tr>
<td>3</td>
<td>Instance I/O Intensive Workload</td>
<td>0.9%</td>
<td>15.8%</td>
</tr>
<tr>
<td>4</td>
<td>Accompanying Slow SQL</td>
<td>8.6%</td>
<td>9.0%</td>
</tr>
<tr>
<td>5</td>
<td>Instance CPU &amp; I/O Intensive Workload</td>
<td>8.1%</td>
<td>4.8%</td>
</tr>
<tr>
<td>6</td>
<td>Host CPU Bottleneck</td>
<td>7.5%</td>
<td>4.1%</td>
</tr>
<tr>
<td>7</td>
<td>Host Network Bottleneck</td>
<td>6.9%</td>
<td>4.1%</td>
</tr>
<tr>
<td>8</td>
<td>External Operations</td>
<td>6.9%</td>
<td>3.5%</td>
</tr>
<tr>
<td>9</td>
<td>Database Internal Problem</td>
<td>3.4%</td>
<td>3.5%</td>
</tr>
<tr>
<td>10</td>
<td>Unknown Problem</td>
<td>2.9%</td>
<td>3.5%</td>
</tr>
</tbody>
</table>

#### End to end performance

![Bar chart showing F1-score comparison between DBSherlock and iSQUAD](chart)

Root causes are not included in DBSherlock [Sigmod 2016]
More in Our Paper

• iSQUAD Efficiency
• BCM Effectiveness
• Parameter Sensitivity
• Contribution of Components
• Multiple Root Causes
• Generality of iSQUAD
• Root Causes to Actions
Database Autonomy Service (DAS)

https://www.alibabacloud.com/help/product/63907.htm
Autonomy Service

Collect Data
- Metrics
- Logs
- Locks
- SQLs
- Tables
- Events

Abnormal Detection
- Timing Anomaly Detection
- Log Anomaly Detection
- Seasonality Identification

Root Cause Analysis
- Abnormal SQL
- Poor Index
- Increased Flow
- Resource Contention
- Poor Parameters
- Slow response clients
- resource bottlenecks

Repair / Optimization
- Automatic SQL Concurrency Limit
- Automatic Index Optimization
- Auto-Scaling
- Parameter Optimization
- Storage Optimization
- Evaluate and deposit in the knowledge base

Feedback
- DAS = Data + Machine Learning + Expert Experience + Automatic Execution

Track Execution Performance
- Find abnormal and roll back automatically
HA iSQ in DAS

- The SQL for the HA (High-availability) of DBs:

```
INSERT INTO X VALUES Y ON DUPLICATE KEY UPDATE id = ${id};
```

### DAS Autonomous Pipeline

- **Perception**: Autonomous Detection/Forecast
- **iSQUAD**: Root Cause Analysis
- **Planning & Action**: Automatic SQL Throttling, Automatic SQL Optimization, Auto-Scaling
- **Knowledge Graph**:...

### Healthy Status

- **T1**: Healthy Status
- **T2**: Potential Problem
- **T3**: Major/Warning/Notified/Minor
- **T4**: Healthy Status

### HA SQL turn to iSQ

< 10s
DAS – Autonomy center

**Exception Analysis**

<table>
<thead>
<tr>
<th>Exception Cause</th>
<th>Category</th>
<th>Related Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU-intensive Workload Metric Changes</td>
<td>Instance CPU Metrics ↑</td>
<td>mysql.cpu_usage ↑ 87.83% ©</td>
</tr>
<tr>
<td></td>
<td>DML-operated Row Metrics ↑</td>
<td>mysql.inodb_rows_inserted ↑ 473.74% ©, mysql.inodb_rows_updated ↑ 307.49% ©</td>
</tr>
<tr>
<td>DML Statement Changes</td>
<td>DML Executions Metrics ↑</td>
<td>mysql.ops ↑ 187.58% ©, mysql.insert.ps ↑ 439.89% ©, mysql.update.ps ↑ 335.24% ©, mysql.insert ↑ 364.78% ©, mysql.select.ps ↑ 43.71% ©</td>
</tr>
<tr>
<td>Thread Surge</td>
<td>Thread Pool Metrics ↑</td>
<td>mysql.active_session ↑ 637.98% ©</td>
</tr>
</tbody>
</table>
DAS – Automatic SQL Throttling

SQL Statements to be Throttled

<table>
<thead>
<tr>
<th>SQL Template</th>
<th>Quantity</th>
<th>Problem Description</th>
<th>Suggestions</th>
<th>Status</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT min(id), max(id) FROM task_event WHERE gmt_modified &lt; ? AND begin_time &gt; ? AND source IN (?) AND id &gt;= ? AND id &lt;= ?</td>
<td>4554</td>
<td>High Resource Consumption and Poor Performance. Duration Percentage: 88.41%</td>
<td>If the SQL statement continues to be submitted, we recommend that you perform SQL throttling.</td>
<td></td>
<td>Enable Throttling</td>
</tr>
</tbody>
</table>

SQL Statements to be Killed
**DAS – Automatic SQL Optimized**

**Problematic SQL Statement**

```sql
SELECT min(id), max(id)
FROM task_event
WHERE gmt_modified < '2020-06-21'
  AND begin_time > '2020-07-09'
  AND source IN (285)
  AND id >= 15973
  AND id <= 8015673
```

<table>
<thead>
<tr>
<th>SQL Template</th>
<th>D6</th>
<th>Executions</th>
<th>Avg Execution Duration (s)</th>
<th>Max Execution Duration (s)</th>
<th>Avg Lock Wait Duration (s)</th>
<th>Max Lock Wait Duration (s)</th>
<th>Avg Scanned Rows</th>
<th>Max Scanned Rows</th>
<th>Avg Returned Rows</th>
<th>Max Returned Rows</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT min(id), max(id) FROM task_event... eno</td>
<td>4886</td>
<td>21.675</td>
<td>64.25</td>
<td>0.018</td>
<td>1.335</td>
<td>999.80K</td>
<td>1000000</td>
<td>1.00</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**SQL Statement Optimization**

- **Diagnostics ID**: sf3536842713dbb8e5e8546
- **Recommended Program**: High
- **Revenue**: $322954.58 Times
- **Execution Status**: Executed

**Index Recommendation**

<table>
<thead>
<tr>
<th>Suggestion Type</th>
<th>Database Name</th>
<th>Suggestion Details</th>
<th>DDL Statement</th>
</tr>
</thead>
</table>
| Create Index    | eno           | Table Name: task_event
Index: idx_source_begin_time(source, begin_time)
ALTER TABLE 'eno', 'task_event' ADD INDEX 'idx_source_begin_time' ('source', 'begin_time') |

---

38
DAS – Auto Scale

Autonomy Center (Documentation) [Current anomaly will be detected within five minutes.]

Type: All (0) Exceptional Events (0) Optimization Events (1) Auto-Scaling Events (1) Others (0)

Last 1 Hour Last 24 Hours Last 2 Days Last 7 Days Aug 8, 2020 12:05:45 - Aug 15, 2020 12:05:45

All events Events with suggestions

Suggestions

Problems and Suggestions

Problems: The instance load is high and the performance is insufficient.

Suggestions: We recommend that you upgrade the specifications.

Specification Recommendation

Original Specifications: 2 Cores, 4G (rds.mysql.s2.large)

Recommended Type: 4 Cores, 8G (rds.mysql.s3.large)
Conclusion

- **Motivation**: identify the problem of Intermittent Slow Queries in cloud databases

- **Challenge**: anomaly diversity, labeling overheads, interpretable

- **Solution**: Anomaly Extraction, Dependency Cleansing, TOPIC, and Bayesian Case Model

- **Deployment**: iSQUAD prototype are used in Alibaba Database
Thank you

yinzheng.yz@alibaba-inc.com
minghua.mmh@alibaba-inc.com
How it works on Aliyun Database Autonomy Service DAS

**Perception Layer**
- Feature Detection
  - Seasonality detection
  - Changepoint detection
  - Trend detection
  - Meanshift detection

**Analysis Layer**
- iSQUAD Clustering Tasks
- Autonomous Forecast Tasks
- Other Plugin Analysis Tasks

**Decision Layer**
- Anomaly Reason Recommendation

**Execution Layer**
- SQL Limit
- SQL Optimize
- AutoScale

**Offline Data**
- Historical TimeSeries Data
- Instance Metrics Data
- SQL Historical Data
- Event Historical Data

**Online Data**
- Streaming TimeSeries Data
- Instance Metrics Data
- SQL Streaming Data
- Event Streaming Data

**Autonomous Forecast (Fit the Best Model)**
- Get the best Model to forecast the next 24 hours
- Statistics Model Sets
- Bayesian Model Sets
- Deep Learning Model Sets
Join us

Email: yinzheng.yz@alibaba-inc.com
阿里云云智能- 数据库自治服务

21届实习生/应届生
社招