



2023 | APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI

2023 · 上海

SkyWalking Summit



纵目



tetrate

演讲主题

2023 APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI



陈修能

江苏纵目信息科技有限公司 创始人

“基于 SkyWalking 实现
全域一体化观测”

目录

CONTENTS

- 01. SkyWalking 可以干哪些事？**
- 02. SkyWalking + ClickHouse**
- 03. 打通 SkyWalking 和 Zabbix**
- 04. Argus 全域监控深度解析**

01

SkyWalking 可以干哪些事？

1. 接入 Prometheus Exporter 取数

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI

OpenTelemetry Metrics Format

The OpenTelemetry receiver supports ingesting agent metrics by meter-system. The OAP can load the configuration at bootstrap. If the new configuration is not well-formed, the OAP may fail to start up. The files are located at `$CLASSPATH/otel-rules`.

Supported handlers:

- `otlp` : [OpenTelemetry](#) gRPC service handler.

Notice: Set `SW_OTEL_RECEIVER=default` through system environment or change `receiver-otel/selector=${SW_OTEL_RECEIVER:default}` to activate the OpenTelemetry receiver.

The rule file should be in YAML format, defined by the scheme described in [MAL](#). Note: `receiver-otel` only supports the `group`, `defaultMetricLevel`, and `metricsRules` nodes of the scheme due to its push mode.

To activate the `otlp` handler and relevant rules of `istio`:

```
receiver-otel:  
  selector: ${SW_OTEL_RECEIVER:default}  
  default:  
    enabledHandlers: ${SW_OTEL_RECEIVER_ENABLED_HANDLERS:"otlp-metrics"}  
    enabledOtelMetricsRules: ${SW_OTEL_RECEIVER_ENABLED_OTEL_METRICS_RULES:"istio-controlplane"}
```

The receiver adds label with key `node_identifier_host_name` to the collected data samples, and its value is from `net.host.name` (or `host.name` for some OTLP versions) resource attributes defined in OpenTelemetry proto, for identification of the metric data.

2. Zabbix Agent Metrics 采集

2023 | APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI

Configuration file

```
# initExp is the expression that initializes the current configuration file
initExp: <string>

# insert metricPrefix into metric name:  <metricPrefix>_<raw_metric_name>
metricPrefix: <string>

# expPrefix is executed before the metrics executes other functions.
expPrefix: <string>

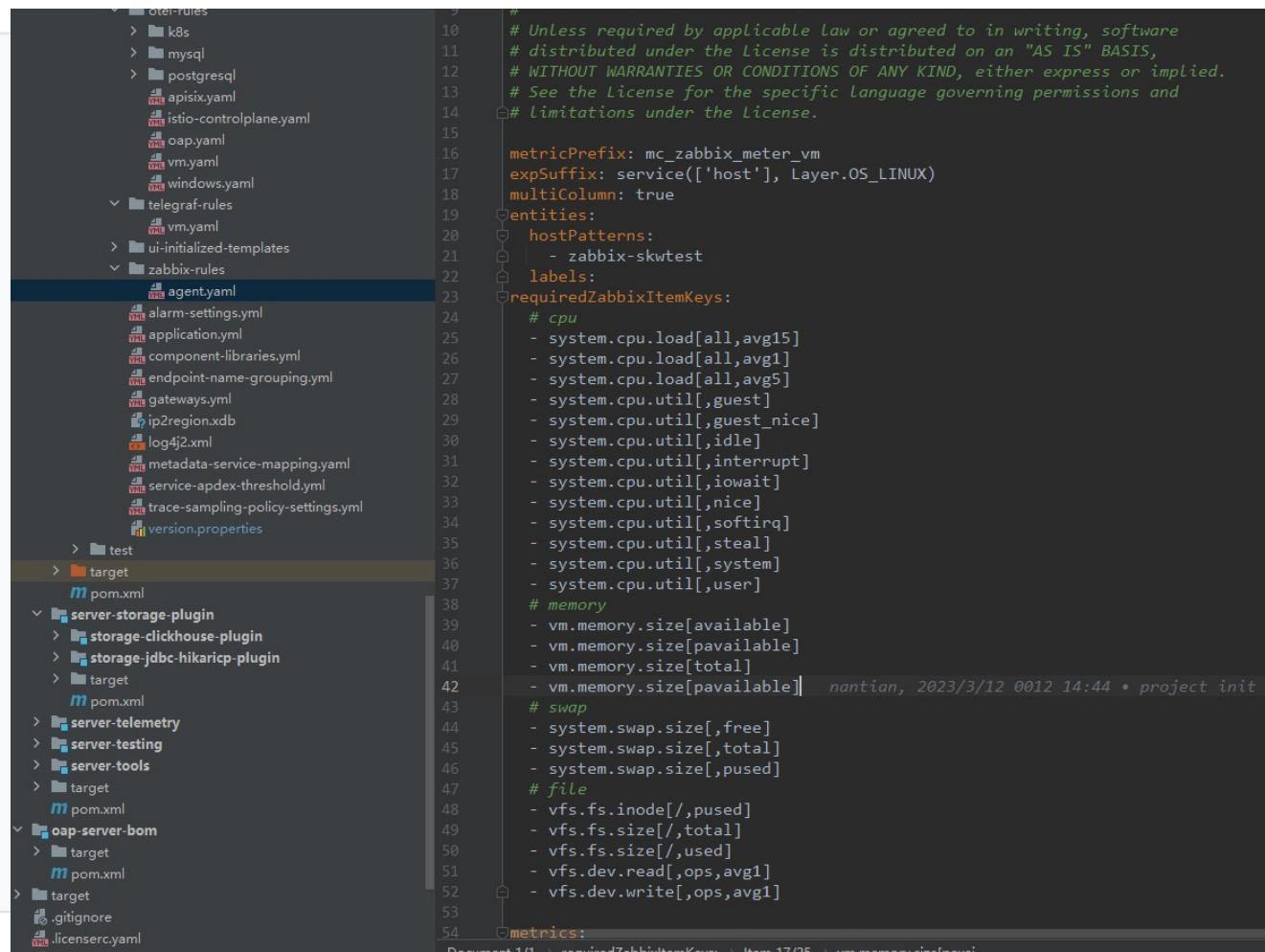
# expSuffix is appended to all expression in this file.
expSuffix: <string>

# Datasource from Zabbix Item keys.
requiredZabbixItemKeys:
  - <zabbix item keys>

# Support agent entities information.
entities:
  # Allow hostname patterns to build metrics.
  hostPatterns:
    - <regex string>

  # Customized metrics Label before parse to meter system.
  labels:
    [- <labels>]

  # Metrics rule allow you to recompute queries.
  metrics:
    [ - <metrics_rules> ]
```



The screenshot shows a code editor with a dark theme displaying a Java project structure. The project includes several modules like 'otel-rules', 'tegraf-rules', 'ui-initialized-templates', 'zabbix-rules', 'test', 'target', 'server-storage-plugin', 'server-telemetry', 'server-testing', 'server-tools', 'oap-server-bom', and 'metrics'. Each module contains various configuration files such as 'apisix.yaml', 'istio-controlplane.yaml', 'oap.yaml', 'vm.yaml', 'windows.yaml', 'agent.yaml', 'alarm-settings.yaml', 'application.yaml', 'component-libraries.yaml', 'endpoint-name-grouping.yaml', 'gateways.yaml', 'ip2region.xdb', 'log4j2.xml', 'metadata-service-mapping.yaml', 'service-apdex-threshold.yaml', 'trace-sampling-policy-settings.yaml', 'version.properties', 'pom.xml', and 'governance.yaml'. The code editor also displays a large block of Java code for the 'Metrics' class, which handles metric prefixing, suffixing, and entity configuration. The code includes annotations like '# Unless required by applicable Law or agreed to in writing, software', 'hostPatterns', 'labels', 'requiredZabbixItemKeys', and various metric definitions for CPU, memory, swap, and file systems.

```
# Unless required by applicable Law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

metricPrefix: mc_zabbix_meter_vm
expSuffix: service(['host'], Layer.OS_LINUX)
multiColumn: true
entities:
  hostPatterns:
    - zabbix-skwtest
  labels:
  requiredZabbixItemKeys:
    # cpu
    - system.cpu.load[,avg15]
    - system.cpu.load[,avg1]
    - system.cpu.load[,avg5]
    - system.cpu.util[,guest]
    - system.cpu.util[,guest_nice]
    - system.cpu.util[,idle]
    - system.cpu.util[,interrupt]
    - system.cpu.util[,iowait]
    - system.cpu.util[,nice]
    - system.cpu.util[,softirq]
    - system.cpu.util[,steal]
    - system.cpu.util[,system]
    - system.cpu.util[,user]
    # memory
    - vm.memory.size[available]
    - vm.memory.size[pavailable]
    - vm.memory.size[total]
    - vm.memory.size[pavailable]
    # swap
    - system.swap.size[,free]
    - system.swap.size[,total]
    - system.swap.size[,pused]
    # file
    - vfs.fs.inode[,pused]
    - vfs.fs.size[,total]
    - vfs.fs.size[,used]
    - vfs.dev.read[,ops,avg1]
    - vfs.dev.write[,ops,avg1]
```

3. 日志数据采集与LAL语言

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI

Enforcer

Enforcer is another special sink that forcibly samples the log. A typical use case of enforcer is when you have configured a sampler and want to save some logs such as to save error logs even if the sampling mechanism has been configured.

```
filter {
    // ... parser

    sink {
        sampler {
            // ... sampler configs
        }
        if (parsed.level == "ERROR" || parsed.userId == "TestingUserId") { // sample error logs or testing users' logs (userId == "TestingUserId")
            enforcer {
            }
        }
    }
}
```

4. SkyWalking OAP 自观测

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI

Telemetry for backend

The OAP backend cluster itself is a distributed streaming process system. To assist the Ops team, we provide the telemetry for the OAP backend itself, also known as self-observability (so11y)

By default, the telemetry is disabled by setting `selector` to `none`, like this:

```
telemetry:  
  selector: ${SW_TELEMETRY:none}  
  none:  
    prometheus:  
      host: ${SW_TELEMETRY_PROMETHEUS_HOST:0.0.0.0}  
      port: ${SW_TELEMETRY_PROMETHEUS_PORT:1234}  
      sslEnabled: ${SW_TELEMETRY_PROMETHEUS_SSL_ENABLED:false}  
      sslKeyPath: ${SW_TELEMETRY_PROMETHEUS_SSL_KEY_PATH:""}  
      sslCertChainPath: ${SW_TELEMETRY_PROMETHEUS_SSL_CERT_CHAIN_PATH:""}
```

You may also set `Prometheus` to enable them. For more information, refer to the details below.

Self Observability

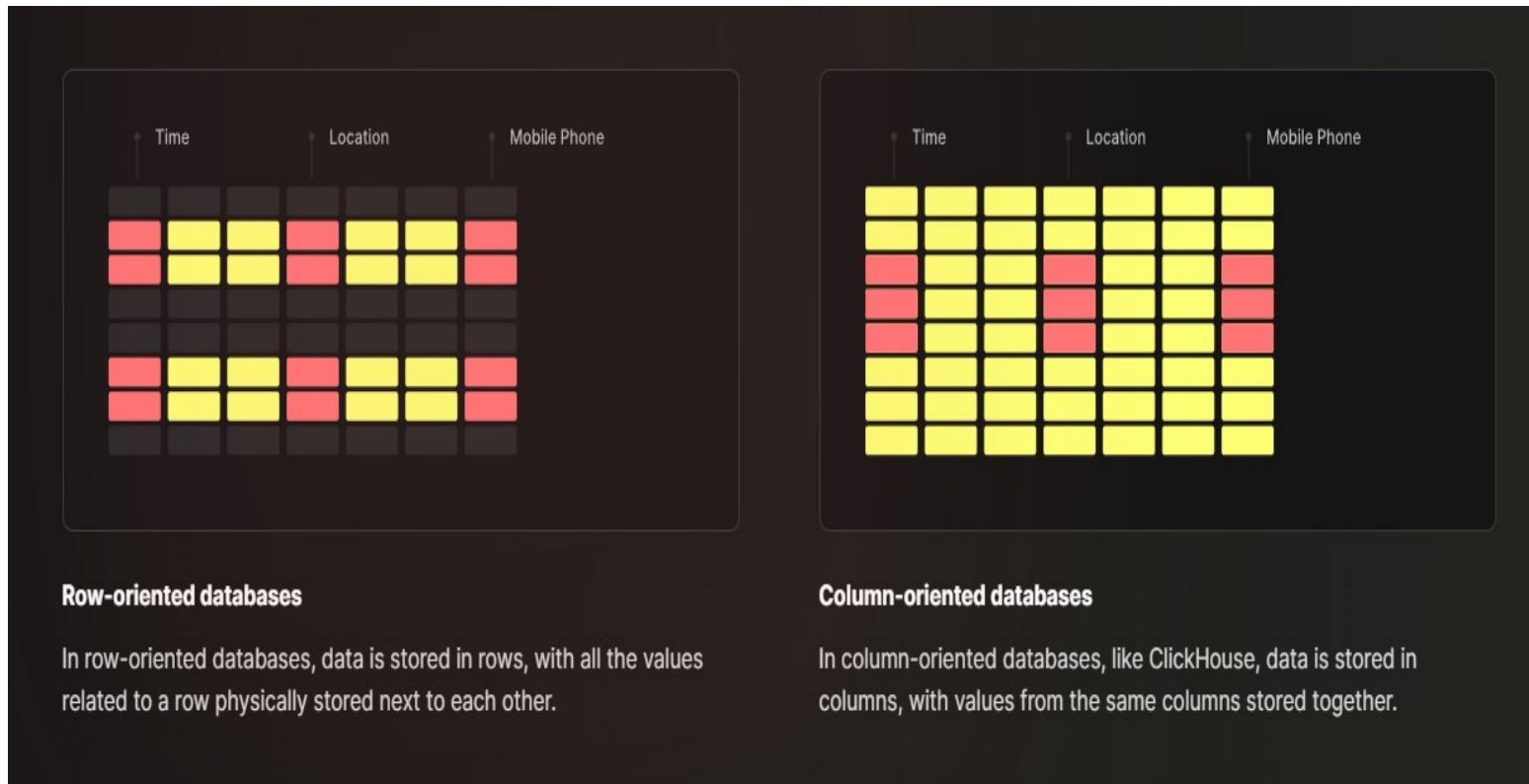
SkyWalking supports exposing telemetry data representing OAP running status through Prometheus endpoint. Users could set up OpenTelemetry collector to scrap and forward telemetry data to OAP server for further analysis, eventually showing up UI or GraphQL API.

02

SkyWalking + ClickHouse

1. 为什么选择 ClickHouse

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI



1 列式存储

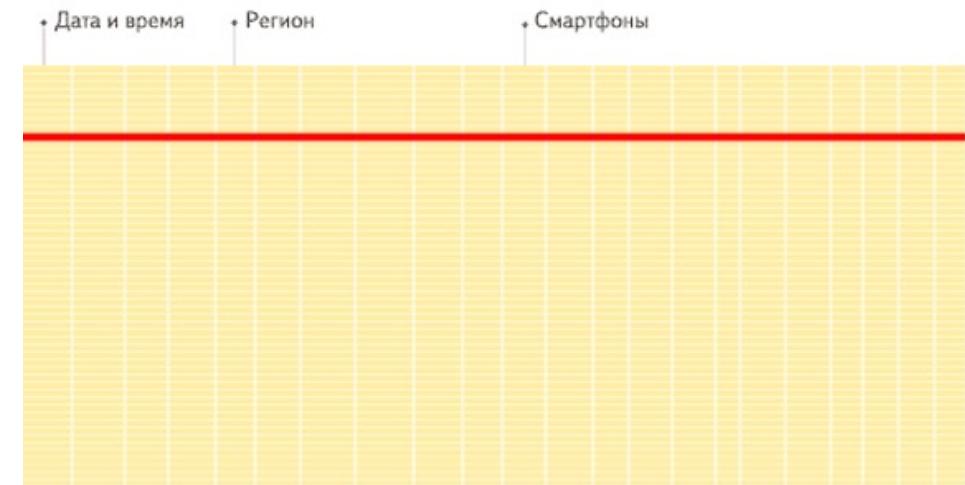
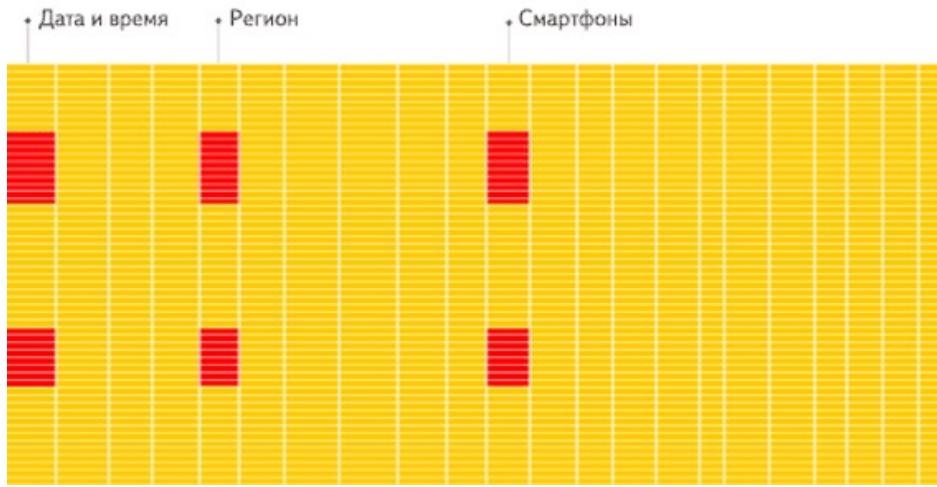
2 高压缩率

3 支持数据合并

4 低基数值的优化

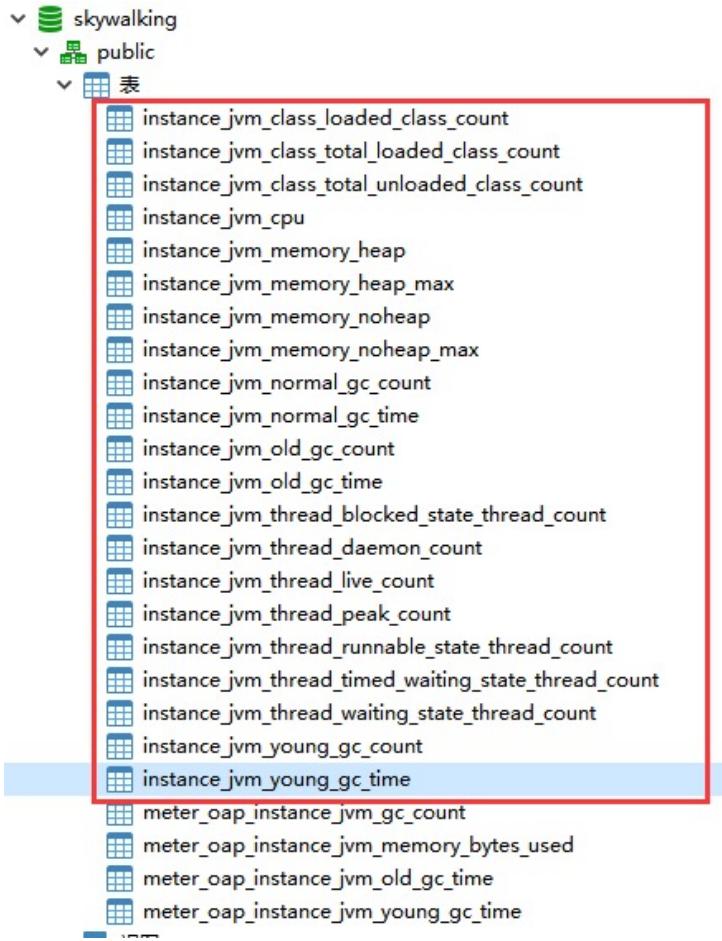
2. 列式存储

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI



3. 基于列式存储重新设计指标存储

2023 APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI



The screenshot shows a database schema for 'skywalking' under 'public'. A red box highlights a group of tables related to 'instance_jvm' metrics:

- instance_jvm_class_loaded_class_count
- instance_jvm_class_total_loaded_class_count
- instance_jvm_class_total_unloaded_class_count
- instance_jvm_cpu
- instance_jvm_memory_heap
- instance_jvm_memory_heap_max
- instance_jvm_memory_noheap
- instance_jvm_memory_noheap_max
- instance_jvm_normal_gc_count
- instance_jvm_normal_gc_time
- instance_jvm_old_gc_count
- instance_jvm_old_gc_time
- instance_jvm_thread_blocked_state_thread_count
- instance_jvm_thread_daemon_count
- instance_jvm_thread_live_count
- instance_jvm_thread_peak_count
- instance_jvm_thread_runnable_state_thread_count
- instance_jvm_thread_timed_waiting_state_thread_count
- instance_jvm_thread_waiting_state_thread_count
- instance_jvm_young_gc_count
- instance_jvm_young_gc_time

Below this group, there are four more tables:

- meter_oap_instance_jvm_gc_count
- meter_oap_instance_jvm_memory_bytes_used
- meter_oap_instance_jvm_old_gc_time
- meter_oap_instance_jvm_young_gc_time

Instance_jvm 一个指标一张表



The screenshot shows a database schema for 'mc.javaagent_instance.jvm' under '列'. A blue box highlights specific columns:

- id (String)
- metric_prefix (LowCardinality(String))
- entity_id (LowCardinality(String))
- service_id (LowCardinality(String))
- class_total_loaded_class_count_value (Int64)
- class_total_loaded_class_count_summation (Int64)
- class_total_unloaded_class_count_value (Int64)
- class_total_unloaded_class_count_summation (Int64)
- class_loaded_class_count_value (Int64)
- class_loaded_class_count_summation (Int64)
- thread_timed_waiting_state_thread_count_value (Int64)
- thread_timed_waiting_state_thread_count_summation (Int64)
- thread_waiting_state_thread_count_value (Int64)
- thread_waiting_state_thread_count_summation (Int64)
- thread_blocked_state_thread_count_value (Int64)
- thread_blocked_state_thread_count_summation (Int64)
- threadRunnable_state_thread_count_value (Int64)
- threadRunnable_state_thread_count_summation (Int64)
- thread_peak_count_value (Int64)
- thread_peak_count_summation (Int64)
- thread_daemon_count_value (Int64)
- thread_daemon_count_summation (Int64)
- thread_live_count_value (Int64)
- thread_live_count_summation (Int64)
- normal_gc_count_value (Int64)
- old_gc_count_value (Int64)
- young_gc_count_value (Int64)
- normal_gc_time_value (Int64)
- old_gc_time_value (Int64)
- young_gc_time_value (Int64)
- memory_noheap_max_value (Int64)
- memory_noheap_max_count (Int64)
- memory_noheap_max_summation (Int64)
- memory_heap_max_value (Int64)
- memory_heap_max_count (Int64)
- memory_heap_max_summation (Int64)
- memory_noheap_value (Int64)
- memory_noheap_count (Int64)
- memory_noheap_summation (Int64)

Instance_jvm 合并一张表

4. 数据合并

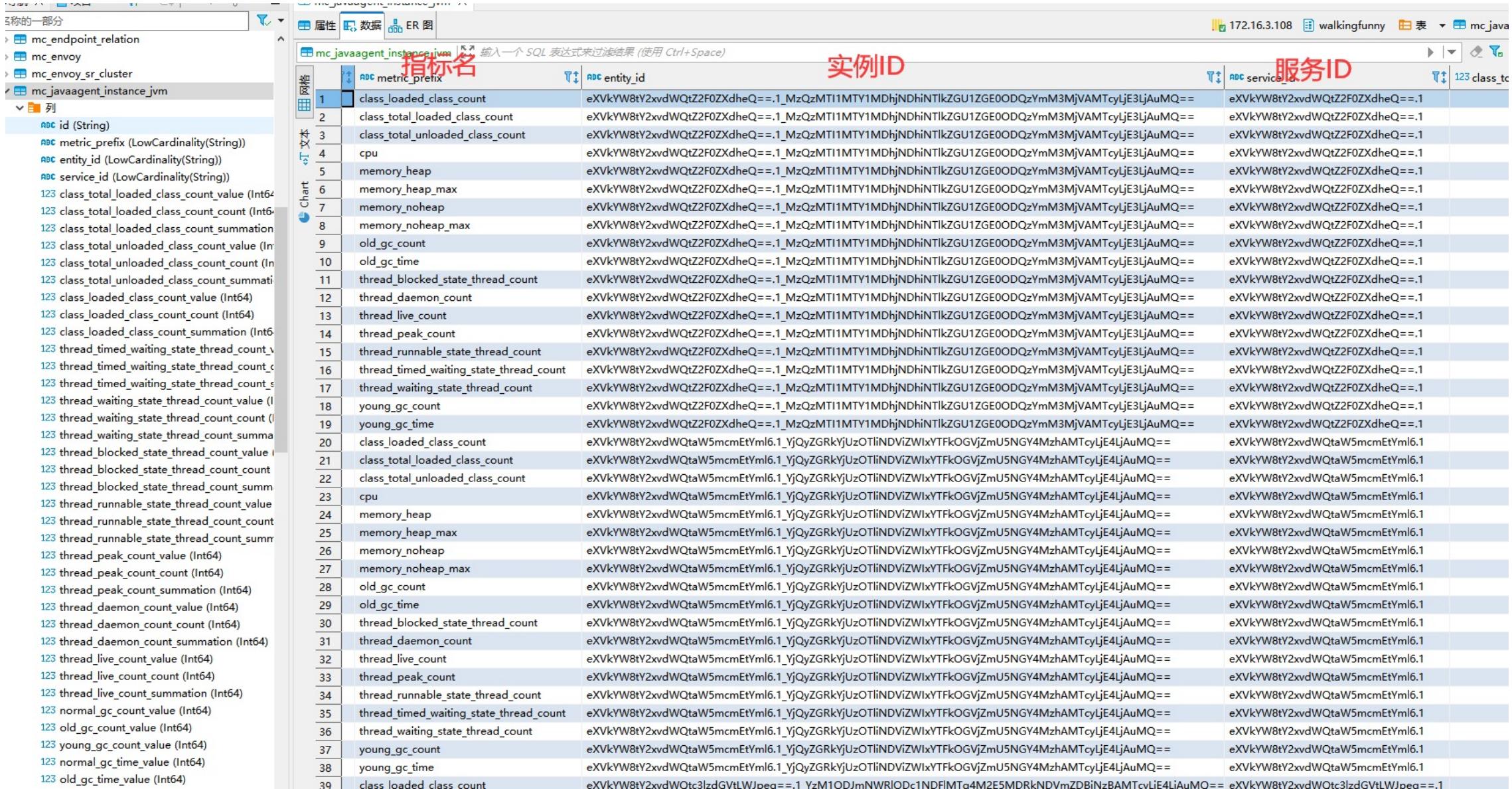
The screenshot displays the Apache SkyWalking UI interface, specifically the 'mc_javaagent_instance_jvm' project. It shows two main tables side-by-side:

- mc_javaagent_instance_jvm** (Left Table):
 - 网格 (Grid):** Shows a list of rows with IDs from 1 to 41. Each row contains a long string of metrics, such as 'abc_id', 'abc_metric_prefix', 'abc_entity_id', etc.
 - 属性 (Properties):** A tree view showing the structure of the metrics, including categories like 'mc_endpoint_relation', 'mc_envoy', 'mc_envoy_sr_cluster', and 'mc_javaagent_instance_jvm'.
- mc_endpoint_relation** (Right Table):
 - 网格 (Grid):** Shows a list of rows with IDs from 1 to 41. Each row contains a long string of metrics, such as 'abc_id', 'abc_metric_prefix', 'abc_entity_id', etc.
 - 属性 (Properties):** A tree view showing the structure of the metrics, including categories like 'mc_endpoint_relation'.

Key features visible in the UI include:

- 搜索 (Search):** A search bar at the top of each table allows users to filter results by SQL expressions.
- 导出 (Export):** Buttons for exporting data to CSV or Excel are located at the top right of each table.
- 指标名 (Metric Name):** A red box highlights the column headers 'abc_id', 'abc_metric_prefix', and 'abc_entity_id' in both tables.
- 唯一ID (Unique ID):** A red box highlights the column 'abc_id' in the left table.

5. 低基数字段优化



The screenshot shows a database interface with a table containing 39 rows of data. The columns are labeled: 指标名 (Metric Name), 实例ID (Instance ID), 服务ID (Service ID), and 123 class_to (Column 123). The first column contains metric names such as 'class_loaded_class_count', 'cpu', 'memory_heap', etc. The second column contains complex SQL queries for each metric. The third column contains shorter service IDs. The fourth column is labeled '123 class_to'.

ABC metric_prefix	实例ID	服务ID	123 class_to
1 class_loaded_class_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
2 class_total_loaded_class_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
3 class_total_unloaded_class_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
4 cpu	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
5 memory_heap	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
6 memory_heap_max	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
7 memory_noheap	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
8 memory_noheap_max	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
9 old_gc_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
10 old_gc_time	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
11 thread_blocked_state_thread_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
12 thread_daemon_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
13 thread_live_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
14 thread_peak_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
15 thread_runnable_state_thread_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
16 thread_timed_waiting_state_thread_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
17 thread_waiting_state_thread_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
18 young_gc_count	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
19 young_gc_time	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1_MzQzMTI1MTY1MDhjNDhiINTlkZGU1ZGE0ODQzYmM3MjVAMTCylE3ljAuMQ==	eXVkyW8tYxvdWQtZ2F0ZXdheQ==.1	
20 class_loaded_class_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
21 class_total_loaded_class_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
22 class_total_unloaded_class_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
23 cpu	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
24 memory_heap	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
25 memory_heap_max	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
26 memory_noheap	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
27 memory_noheap_max	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
28 old_gc_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
29 old_gc_time	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
30 thread_blocked_state_thread_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
31 thread_daemon_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
32 thread_live_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
33 thread_peak_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
34 thread_runnable_state_thread_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
35 thread_timed_waiting_state_thread_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
36 thread_waiting_state_thread_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
37 young_gc_count	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
38 young_gc_time	eXVkyW8tYxvdWQtA5mcmEtYml6.1_YjQzGRkYjUzOTiNDViZWIxYTFkOGVjZmU5NGY4MzhAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtA5mcmEtYml6.1	
39 class_loaded_class_count	eXVkyW8tYxvdWQtC3zdGvtLWpea==.1_Yz1ODjnNWRlODc1NDFlMTa4M2E5MDRkNDVmZD8iBzAMTCylE4ljAuMQ==	eXVkyW8tYxvdWQtC3zdGvtLWpea==.1	

03

打通 SkyWalking + Zabbix

1. Zabbix对Java等中间件监控的痛点

2023 | APACHE • SkyWalking
SUMMIT | CHINA · SHANGHAI

- 1 Zabbix 监控 jvm 的信息必须要通过 JMX 协议来获取，这个强依赖JavaGateway 网关
- 2 对于多个 JVM 实例监控的配置比较麻烦
- 3 JVM 信息采集的不够全，很多涉及到接口性能的数据获取不到

2. Zabbix 基于PromQL采集指标数据

Item Tags Preprocessing

* Name

Type **HTTP agent**

* Key Select

Type of information Numeric (unsigned)

* URL Parse

Query fields

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> Remove

Add

Request type GET

* Timeout 3s

Request body type Raw data

Request body

Headers

Name	Value
<input type="text" value="name"/>	<input type="text" value="value"/> Remove

Add

Required status codes 200

Follow redirects

Retrieve mode Body

Convert to JSON

HTTP agent

04

Argus 全域监控深度解析

Zabbix Plus = ArgusOMS

SkyWalking 二开 = ArgusAPM

1. Argus 概览页

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI

argus

监控摘要 / 首页

所有当前告警>>

告警列表

紧急 0 高级 8 中级 3 低级 0 提示 0

2021-01-09 15:31:19 【高级】防火墙 USG6300E 温度在最近5分钟>60°C

2021-01-09 13:17:45 【中级】三石Jenkins http 服务运行状态在最近1分钟正常

2021-01-09 11:28:34 【中级】Dell PowerEdge R730xd PowerEdge R730xd CPU2 Temp: Temperature在最近1分钟>60°C

2021-01-08 10:15:01 【中级】大哥Host Windows 系列 SNMP可用性在最近1分钟正常

资源监控详情

操作系统 2 监控总数

数据库 2 标准应用

服务器 已启用

网络设备 0

中间件 已禁用

虚拟化 虚拟化

出入口流量

移动 带宽: 200Mbps 0 866 流出 Kbps

电信 带宽: 300MB 0 452 流出 Kbps

0 171 流入 Kbps 1544 流入 Kbps

主机运行 Top5

主机流量 单位Kbps

主机名	流量 (Kbps)
智远host	755.18
三石Host	742.21
ArgusDe...	654.99
NAS	221.6
mysql.z...	197.99

主机CPU负载

主机名	CPU 使用率 (%)
155-doc...	100%
三石Host	77%
mysql.z...	25%
ArgusDe...	24.75%
159-doc...	7.25%

内存使用率

主机名	内存使用率 (%)
154-gate...	87.57%
zmops-vi...	84.75%
智远host	65.00%
三石Host	64.00%
148-ubun...	61.48%

网络设备运行 Top5

网络设备流量 单位Mbps

设备名	流量 (Mbps)
导入交换机	5.8
sw1143	5.43
sw1132	5.42
H3C131	4.35
防火墙	3.06

网络设备CPU负载

设备名	CPU 使用率 (%)
导入交换机	56.00%
sw1132	56.00%
sw1143	56.00%
防火墙	14.00%

网络设备内存负载

设备名	内存使用率 (%)
防火墙	71%
导入交换机	27%
sw1132	27%
sw1143	27%

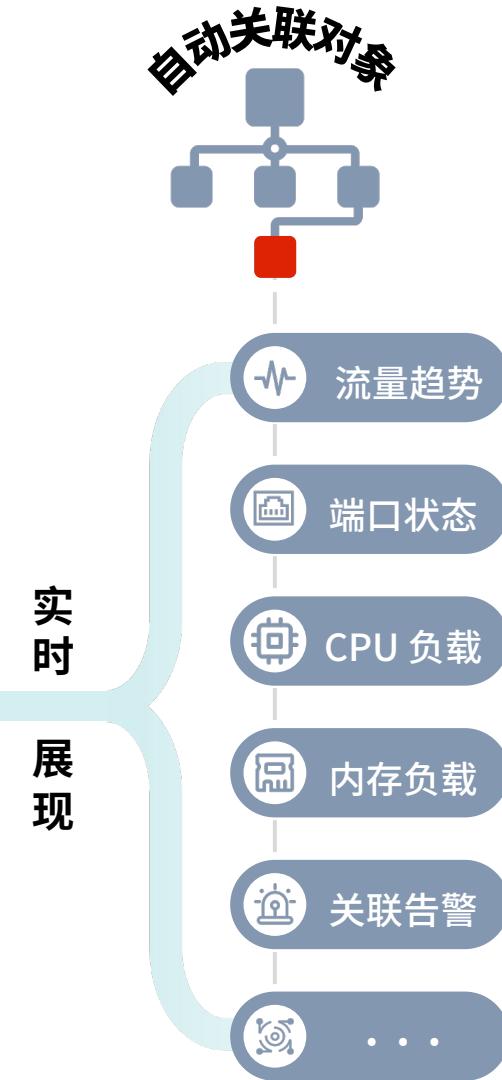
2. 全自动的网络动态拓扑

全自动生成网络拓扑



The screenshot shows the argus network monitoring interface. At the top, there's a navigation bar with links like '首页', '监控大屏', '生成树', 'CDP', 'LLDP', 'OSPF', 'ISIS', 'ARP', and '...'. The main area features a large network topology graph with nodes representing different network devices. Labels for network segments include '核心业务系统', '区域办事处', 'OA', '生产网络', '无线网', '实验室', and '对外门户'. Each node has its IP address and specific details like 'vCenter' or 'Apache' visible. To the right of the graph, there are several summary statistics and charts:

- 设备总数:** 82 网络设备, 51 服务器
- 设备告警总数:** 1 网络设备, 5 服务器
- 警告统计:** 显示了不同级别的告警数量，如严重、高危等。
- 最近告警设备:** 列出了最近有告警的设备IP地址。



3. 服务主动异地拨测

全部拨测

状态速览

正常	故障	暂停
3	7	1

状态历史

名称	状态	日期时间	消息
拨测-win3.25	故障	2022-11-18 16:15:33	
拨测-linux2.147	故障	2022-11-18 16:11:30	
拨测-linux40.145	故障	2022-11-18 16:10:22	
拨测-ND03	正常	2022-11-18 15:55:13	
拨测-ND02	正常	2022-11-18 15:53:48	
拨测-AC	正常	2022-11-18 15:51:54	
拨测PING-3.184	故障	2022-11-18 14:55:06	
拨测PING-2.147	故障	2022-11-18 14:51:42	
拨测PING-3.132	故障	2022-11-18 14:14:00	
拨测-三石web服务	暂停	2022-11-02 18:28:04	
http 服务拨测	故障	2022-10-21 10:27:56	

共 11 条 < > 前往 页

全部拨测 < > 前往 页

4. Syslog 分析

2023 APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI



4. Syslog 分析

基础信息

名称

原始日志

_host

level

service

source

_rawMessage

请先输入原始日志，以获取解析字段

自动获取

手动输入

前序步骤解析结果(自动生成)



定义解析规则

类型

字段

错误处理

高级选项

解析结果



定义解析规则

类型

GROK 解析



字段

_rawMessage



GROK 表达式

请输入GROK表达式

0

错误处理

跳过当前算子逻辑



高级选项

5. 设备配置文件备份

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI

The screenshot shows the argus network management interface. On the left is a dark sidebar with various monitoring and management modules. The main area is titled "网络管理 / 网络设备". It features a search bar with filters for type, brand, location, IP address, and object name. Below the search bar is a table listing network devices. Two rows are visible: one for a "纵目核心交换机" (Zongmu Core Switch) and another for a "纵目出口设备" (Zongmu Export Device). Both entries show their IP addresses, management levels, brands, unique codes, models, operating times, CPU usage, memory usage, monitoring methods, alert rules, operators, and backup status. A modal window titled "已备份配置文件" (Backup Configuration Files) is displayed in the center. It lists two backup entries with columns for backup time, backuper, file name, and operations (Download, Delete). The bottom of the modal and the main page both have pagination controls.

对象名称	IP地址	管理等级	品牌	唯一识别码	型号	运行时长	CPU使用率(%)	内存使用率(%)	监控方式	告警规则	运维人员	备份	操作
纵目核心交换机	172.1.1.4	二级	华为	21980108522S...	S5720SV2-28P-LI-AC	66天,05:09:54	6	27	SNMPv2	Z	超级管...	2	命令行
纵目出口设备	172.1.1.1	二级	华为	10199...	USG6331E	66天,05:08:16	7	71	SNMPv2	Z	超级管...	0	命令行

备份时间	备份人	文件名称	操作
2023-11-02 13:48:05	超级管理员	20231102134805_...	<button>下载</button> <button>删除</button>
2023-11-02 13:47:54	超级管理员	20231102134749_...	<button>下载</button> <button>删除</button>

6. 原生内置了200+模板，开箱即用

The screenshot shows the argus monitoring platform's '资源配置 / 添加资源' (Resource Configuration / Add Resource) page. The left sidebar has a dark theme with various monitoring modules like Monitoring Summary, Network Management, Host Application, Database Management, Hardware Management, Virtualization, IOT Terminal, Topology Management, Alert Management, Special Monitoring, Auto Discovery, Statistics Reports, Basic Settings, and Resource Configuration. Under Resource Configuration, there are sections for Adding Resources, Resource Comprehensive Management, Network Devices, Wireless AC Devices, Wireless AP Devices, Server Hardware, Operating Systems, Databases, Intermediary Components, Standard Applications, and Terminal Devices.

The main content area shows three steps: '型号/版本选择' (Model/Version Selection), '监控方式配置' (Monitoring Method Configuration), and '完善运维信息' (Complete Operation and Maintenance Information). Below these steps is a grid of resource categories: 网络设备 (Network Device), 无线设备 (Wireless Device), 服务器 (Server), 存储设备 (Storage Device), 操作系统 (Operating System), 虚拟化 (Virtualization), 数据库 (Database), 中间件 (Middleware), 标准应用 (Standard Application), and 终端设备 (Terminal Device). The '网络设备' category is highlighted with a red border.

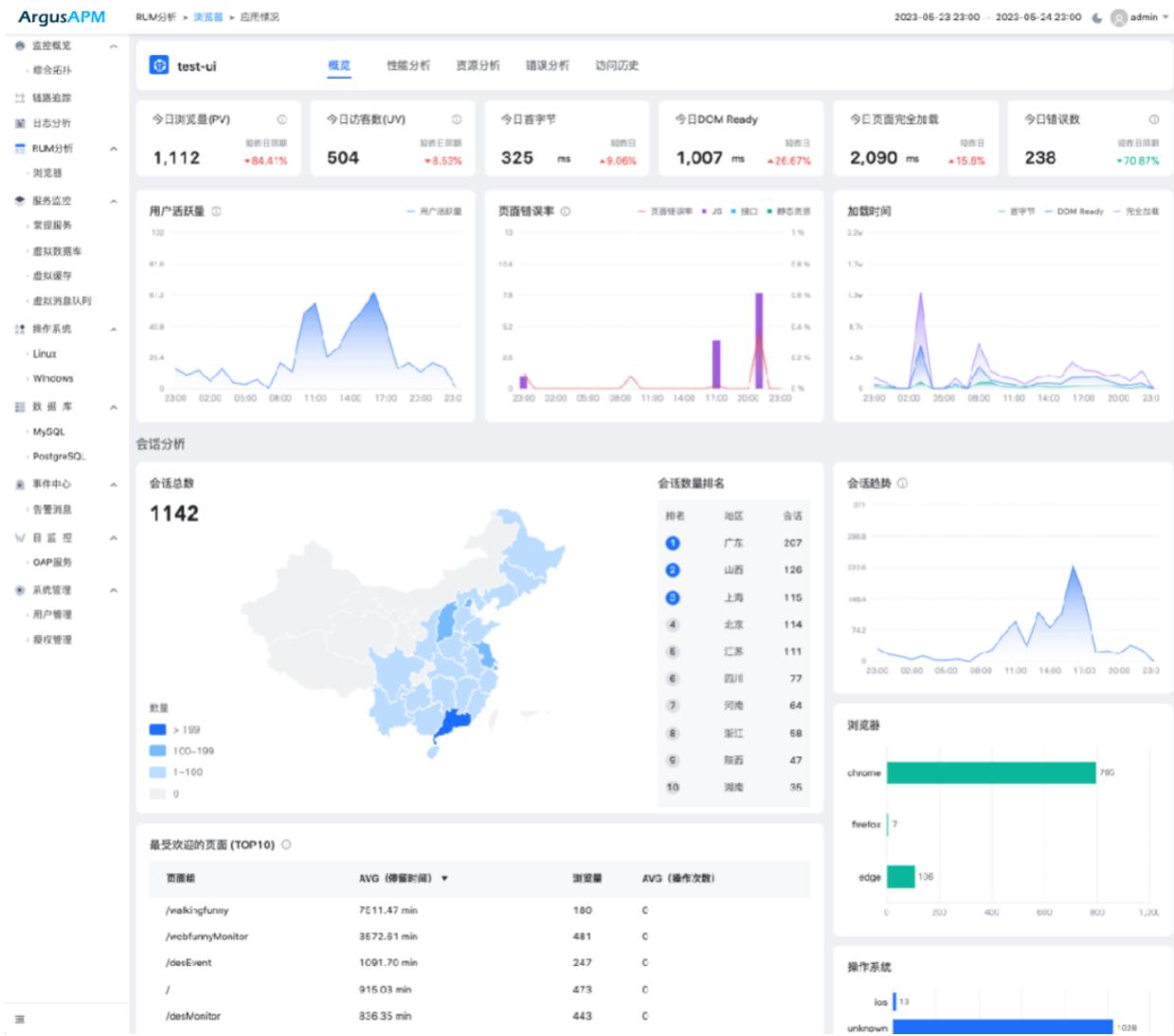
A search bar at the bottom left says '搜索型号名或过滤结果...' (Search model name or filter results...). Below it is a table listing 10 network device models:

选择	序号	型号	品牌	类型	系列
<input type="radio"/>	1	H3C E528	H3C	交换机	E552/528
<input type="radio"/>	2	H3C E552	H3C	交换机	E552/528
<input type="radio"/>	3	10508V	H3C	交换机	H3C 10500
<input type="radio"/>	4	10512	H3C	交换机	H3C 10500
<input type="radio"/>	5	10504	H3C	交换机	H3C 10500
<input type="radio"/>	6	10508	H3C	交换机	H3C 10500
<input type="radio"/>	7	1910-24G	H3C	交换机	H3C 1900
<input type="radio"/>	8	1910-24G-PoE-365W	H3C	交换机	H3C 1900
<input type="radio"/>	9	1910-8-PoE+	H3C	交换机	H3C 1900
<input type="radio"/>	10	1910-16G	H3C	交换机	H3C 1900

At the bottom, there are buttons for '共 2374 条' (Total 2374 items), '25条/页' (25 items per page), and a navigation bar with pages 1 through 95. A 'Next Step' button is at the bottom right.

7. 端侧的用户访问体验数据

2023 APACHE · SkyWalking
CHINA · SHANGHAI



8. 端侧错误统计分析

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI

数据分析精准定位到特定用户，真正做到用户体验分析



概览 错误列表 版本分析 待处理错误 处理概览

全部错误 ▾

错误聚合归类

状态 报错趋势: 30min 发生次数 影响人数 处理人

错误类型	状态	报错趋势	发生次数	影响人数	处理人
CustomizeError	未解决	14天前	1.2千	10	超级管理员
CustomizeError	未分配	40	3		
CustomizeError	未分配	32	9		
CustomizeError	未分配	32	9		
Unknown	未分配	30	11		
UncaughtInPromiseError	未分配	19	5		
CustomizeError	未分配	17	3		
CustomizeError	已解决	8	5	3	超级管理员
CustomizeError	未分配	5			

导出 排序 发生次数 今天

反向源码定位

TypeError http://www.webfunny.cn/webfunny/js/httpError.73361c41043efcf5d27a.min.js:1:20979
Cannot read properties of undefined (reading 'O')
① 今天 2022-08-20 11:48:35

解决 忽略 行为轨迹

24e498... chrome 版本: 103.0.0 PC

数据概况

8 生产总次数 2 影响用户数

Android 0次 Apple 0次 8次

报错趋势

24小时内(点击查看其它时段)

Js错误堆栈 如何使用?

Cannot read properties of undefined (reading 'O')

http://www.webfunny.cn/webfunny/js/httpError.73361c41043efcf5d27a.min.js:1:20979

【堆栈代码位置】: test.endDate||scope;c=;function(e){for(t=e.data,n=e.startDate,r=[],l=0,s=0;s<=c;s+=)(r.push(t[s].day),l.push(t[s].count);var i=n[s].count>100?100:n[s].count;o.push(l))}o.setStatute({ht

【源码解析位置】: 解析源码

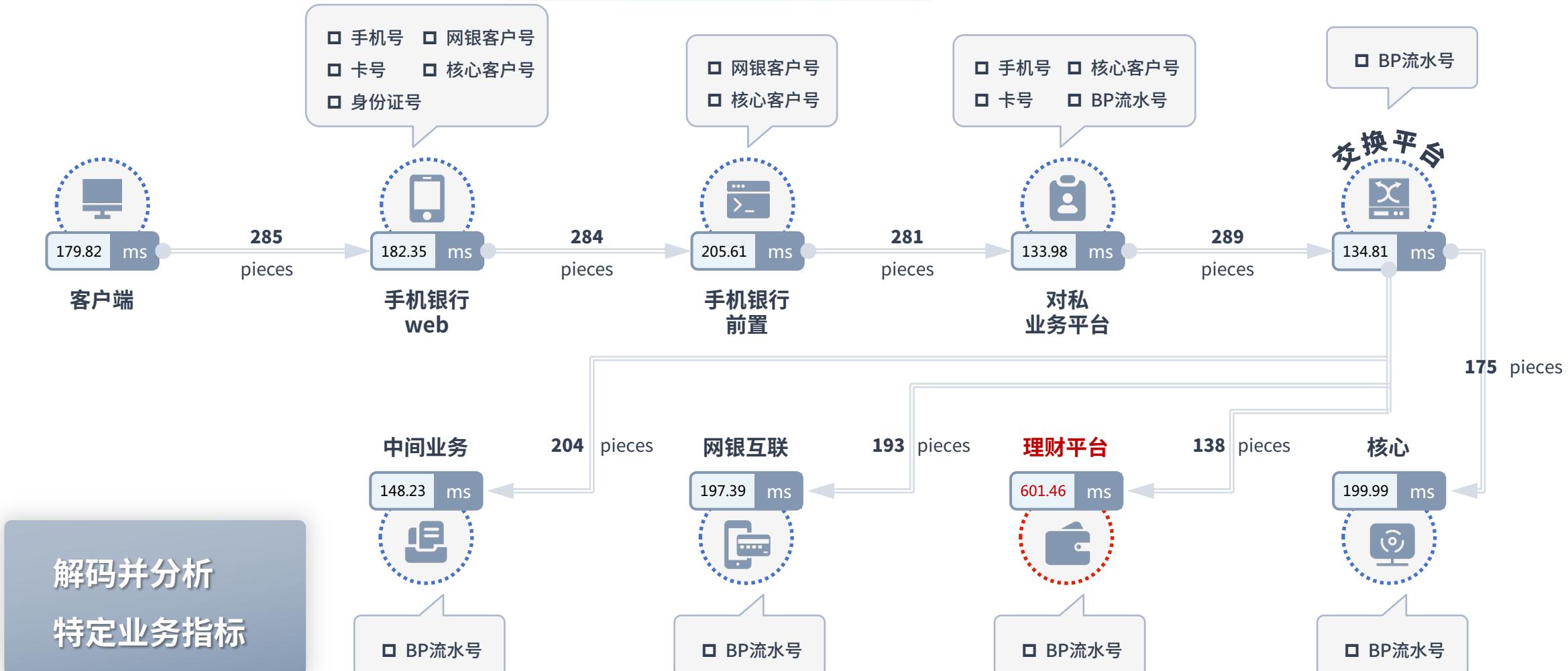
http://www.webfunny.cn/webfunny/js/httpError.73361c41043efcf5d27a.min.js:1:21172

堆栈明细

TypeError: Cannot read properties of undefined (reading 'O') at t (http://www.webfunny.cn/webfunny/js/httpError.73361c41043efcf5d27a.min.js:1:20979)
http://www.webfunny.cn/webfunny/js/httpError.73361c41043efcf5d27a.min.js:1:21172

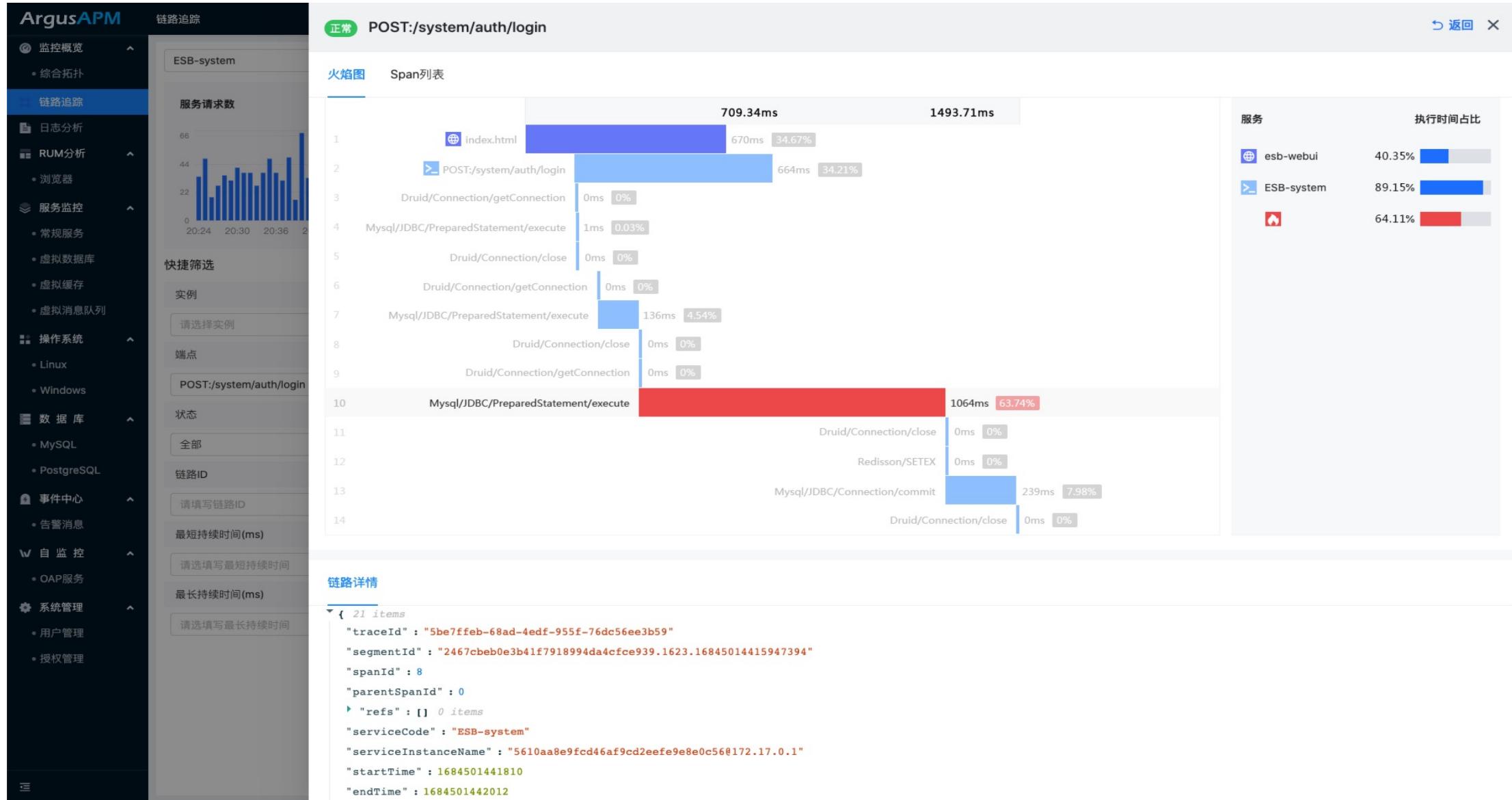
9. 业务链路 - 业务流探针自定义开发

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI



10. 增强了链路追踪的分析

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI



11. 新增 ClickHouse 存储

2023 APACHE · SkyWalking
SUMMIT CHINA · SHANGHAI

数据优势



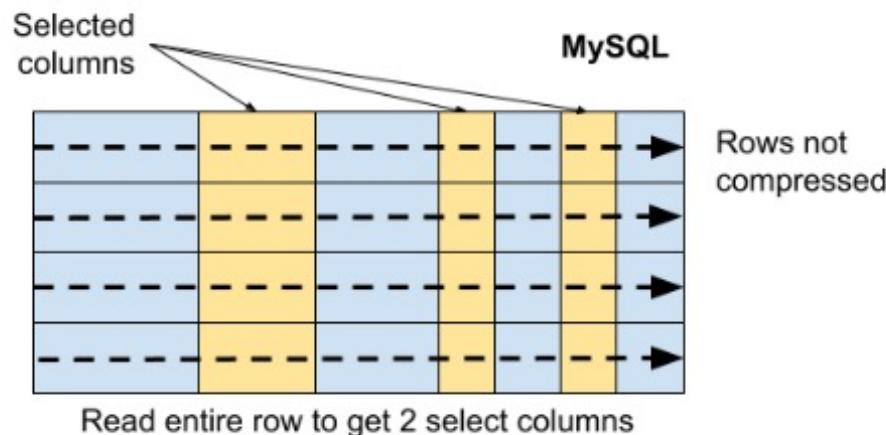
Batch 写入速率提高近 **一倍**，查询延迟全部在 **毫秒级**



单表多列进行压缩后，相较于 ES，存储成本仅需原本的 **10%**



在同等环境下，机器规模可以节省将近 **50%**



Q&A

欢迎提问交流
(仅限2位提问)



南天
中国大陆 上海



扫一扫上面的二维码图案，加我为朋友。



2023 | APACHE • SkyWalking
SUMMIT CHINA · SHANGHAI

2023 • SkyWalking Summit

感谢您的观看



纵目



tetrate