



# 容器化应用痛点剖析：问题诊断、监控及运维

莫源

[zhongwei.lzw@alibaba-inc.com](mailto:zhongwei.lzw@alibaba-inc.com)



扫码关注【阿里巴巴云原生】公众号  
获取第一手技术干货

# 问题定义 – 常见的问题分类与总结

## 影响Kubernetes中对象生命周期的

- 虚拟路由条目无法添加，造成Node不Ready。
- Node资源不足触发Pod驱逐。
- Pod OOM或者Panic进行重启。
- CNI插件异常造成sandbox无法启动。
- HPA无法获取监控数据，造成无法伸缩。
- Autoscaler无法伸缩节点。

通常这类问题在Kubernetes可以通过`describe`资源对象的方式进行排查，通常事件会提供第一手的排查线索，再根据经验定位问题本因和瓶颈。

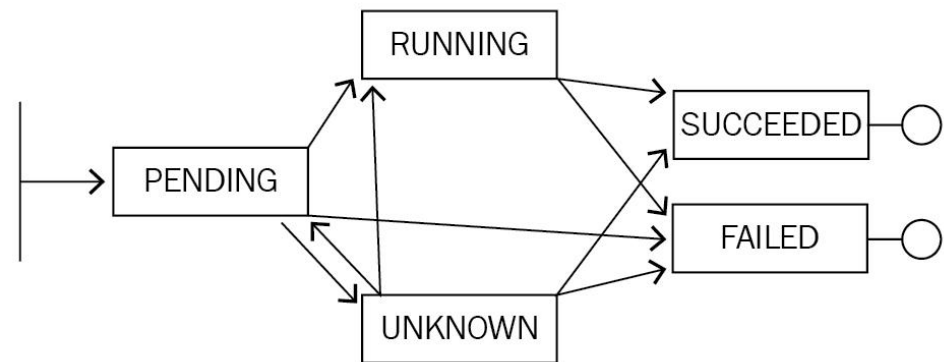
## 不影响Kubernetes中对象生命周期的

- 链路层上超时、网络不通等问题。
- 资源层监控指标毛刺、资源用量异常等。
- 接入层出现异常请求访问状态码、日志。
- 应用层出现请求处理异常日志等。
- 中间件出现RT飙高、IOPS飙满等。

通常这类问题在Kubernetes侧无法发现异常，需要依赖外部系统进行监控和诊断。

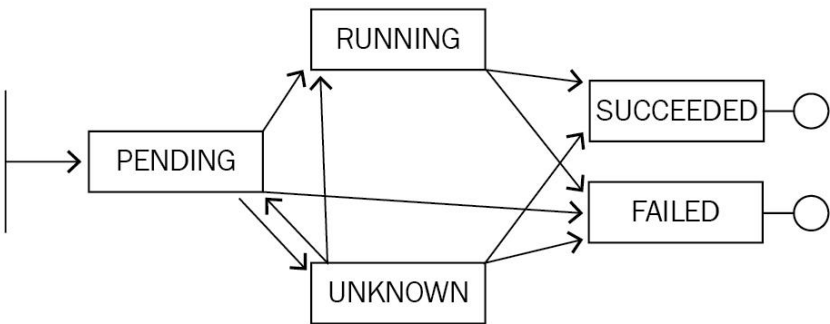
# 问题排查 – 不同问题的定位方式

- 影响Kubernetes中对象生命周期的
  - 事件监控
- 不影响Kubernetes中对象生命周期的
  - 资源监控
  - 应用监控
  - 性能监控
  - 安全监控
  - 日志监控
  - 其他场景化、定制化监控



Pod生命周期示意图

# 状态转换 – Kubernetes生命周期管理的核心



Pod生命周期示意图

Value	Description
Pending	The Pod has been accepted by the Kubernetes system, but one or more of the Container images has not been created. This includes time before being scheduled as well as time spent downloading images over the network, which could take a while.
Running	The Pod has been bound to a node, and all of the Containers have been created. At least one Container is still running, or is in the process of starting or restarting.
Succeeded	All Containers in the Pod have terminated in success, and will not be restarted.
Failed	All Containers in the Pod have terminated, and at least one Container has terminated in failure. That is, the Container either exited with non-zero status or was terminated by the system.
Unknown	For some reason the state of the Pod could not be obtained, typically due to an error in communicating with the host of the Pod.

```
Name: nginx-deployment-basic-c4598964d-tjjnn
Namespace: default
Priority: 0
PriorityClassName: <none>
Node: cn-beijing.192.168.3.167/192.168.3.167
Start Time: Tue, 02 Jul 2019 10:00:40 +0800
Labels: app=nginx
        pod-template-hash=c4598964d
Annotations: <none>
Status: Pending
IP: 172.20.1.214
Controlled By: ReplicaSet/nginx-deployment-basic-c4598964d
Containers:
  nginx:
    Container ID:
    Image: nginx:ne
    Image ID:
    Port: 80/TCP
    Host Port: 0/TCP
    State: Waiting
      Reason: ImagePullBackOff
    Ready: False
    Restart Count: 0
    Environment: <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-nrmg8 (ro)
Conditions:
  Type           Status
  Initialized    True
  Ready          False
  ContainersReady False
  PodScheduled   True
Volumes:
  default-token-nrmg8:
    Type: Secret (a volume populated by a Secret)
    SecretName: default-token-nrmg8
    Optional: false
QoS Class: BestEffort
Node-Selectors: <none>
Tolerations: node.kubernetes.io/not-ready:NoExecute for 300s
             node.kubernetes.io/unreachable:NoExecute for 300s
Events:
  Type     Reason      Age    From          Message
  ----     -
  Normal   Scheduled   37s    default-scheduler   Successfully assigned default/nginx-deployment-basic-c4598964d-tjjnn to cn-beijing.192.168.3.167
  Normal   BackOff     31s    kubelet, cn-beijing.192.168.3.167   Back-off pulling image "nginx:ne"
  Warning  Failed      31s    kubelet, cn-beijing.192.168.3.167   Error: ImagePullBackOff
  Normal   Pulling     16s (x2 over 36s)  kubelet, cn-beijing.192.168.3.167   pulling image "nginx:ne"
  Warning  Failed      10s (x2 over 31s)  kubelet, cn-beijing.192.168.3.167   Failed to pull image "nginx:ne": rpc error: code = Unknown desc = Error response from daemon: manifest f
  Warning  Failed      10s (x2 over 31s)  kubelet, cn-beijing.192.168.3.167   Error: ErrImagePull
```

# 常见Pod的异常与分析

## Pod停留在Pending

Pending表示调度器没有介入，可以通过`kubectl describe pod`，查看事件排查，通常和资源使用相关。

## Pod停留在waiting

一般表示Pod的镜像没有正常的拉取，通常可能和私有镜像拉取，镜像地址不存在，镜像公网拉取相关。

## Pod不断被拉起且可以看到crashing

通常表示Pod已经完成调度并启动，但是启动失败，通常是由于配置、权限造成，需查看Pod日志。

## Pod处在Running但是没有正常工作

通常是由于部分字段拼写错误造成的，可以通过校验部署来排查，例如：`kubectl apply --validate -f pod.yaml`

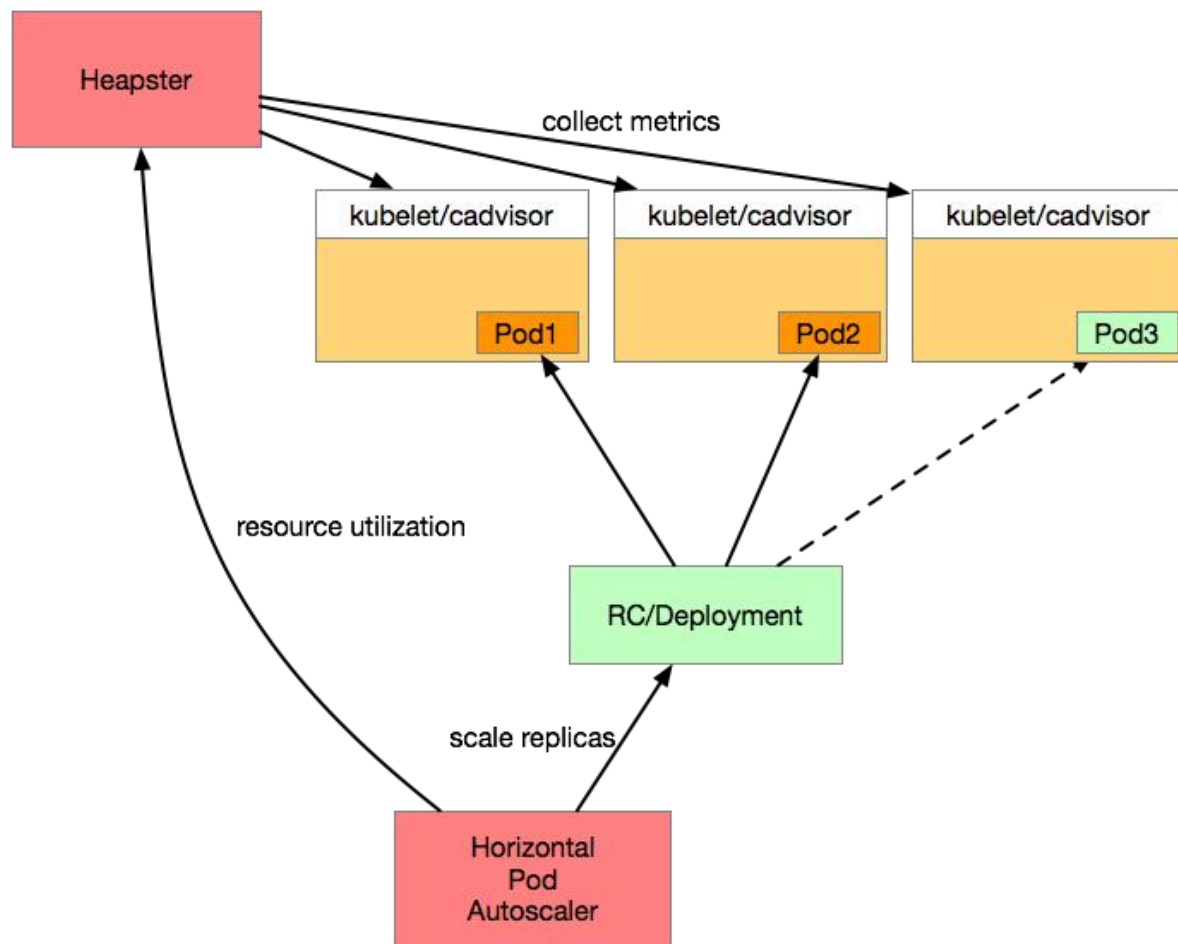
## Service无法正常工作

在排除网络插件自身的问题外，最可能的是label配置有问题，可以通过查看endpoint的方式进行检查。

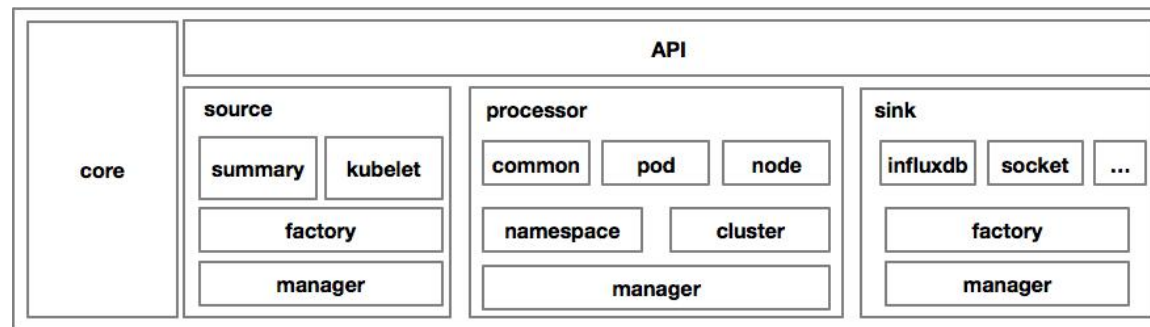
<https://github.com/AliyunContainerService/kubernetes-ops-handbook>



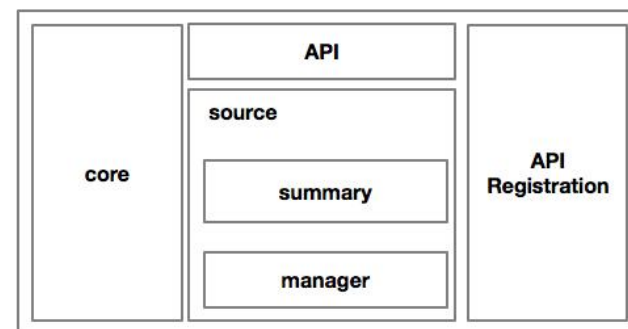
# Kubernetes生态下的监控类组件 – 内置的Metrics-Server



## Heapster体系架构



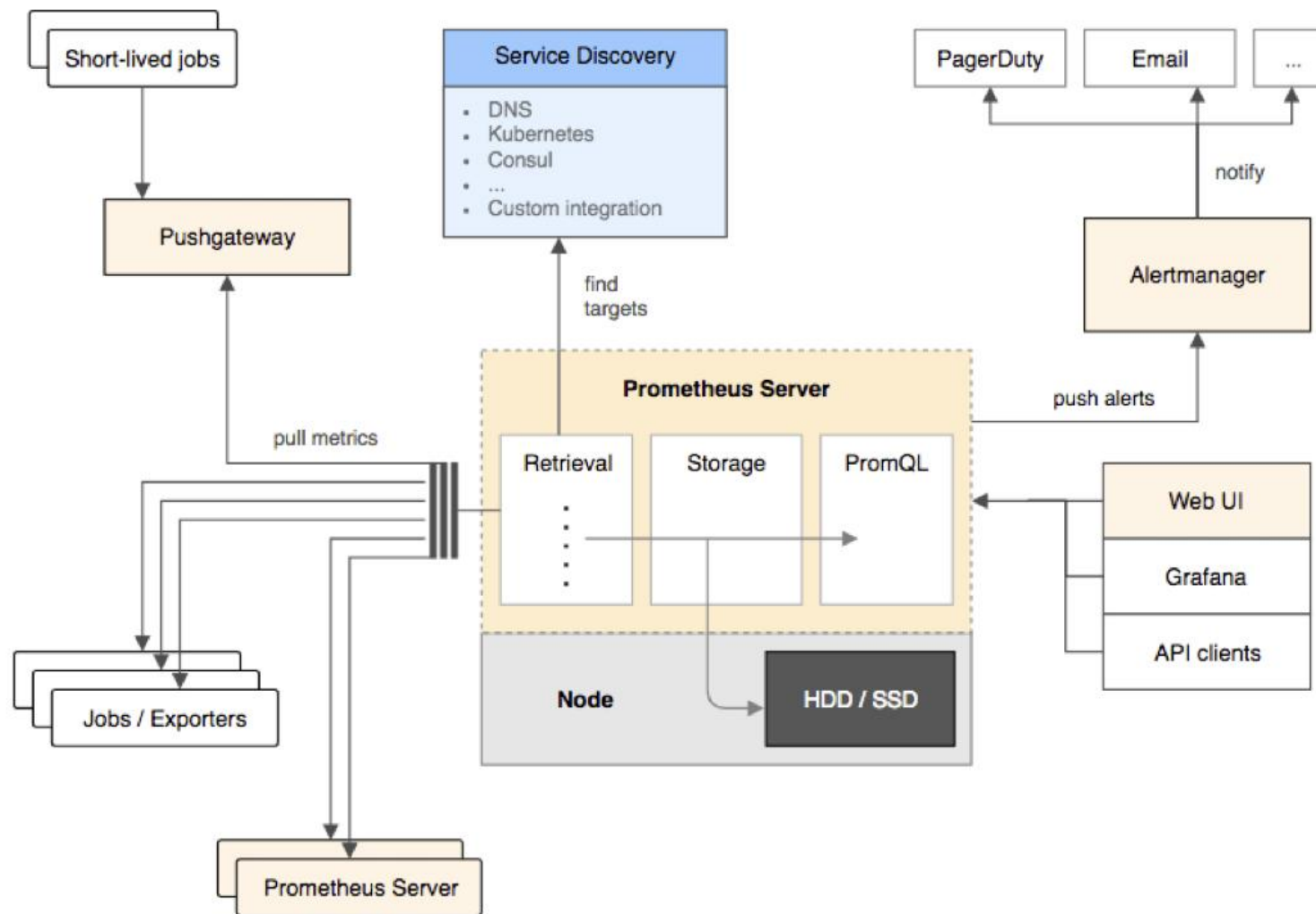
## Metrics-Server体系架构



# Kubernetes监控的演进 – API的标准化

	API	注释
Resource Metrics	<a href="https://metrics.k8s.io">metrics.k8s.io</a>	主要的实现为Metrics-Server，提供资源监控
Custom Metrics	<a href="https://custom.metrics.k8s.io">custom.metrics.k8s.io</a>	主要的实现为Prometheus，提供资源监控和自定义监控
External Metrics	<a href="https://external.metrics.k8s.io">external.metrics.k8s.io</a>	主要的实现为云厂商的Provider，提供云资源的监控指标

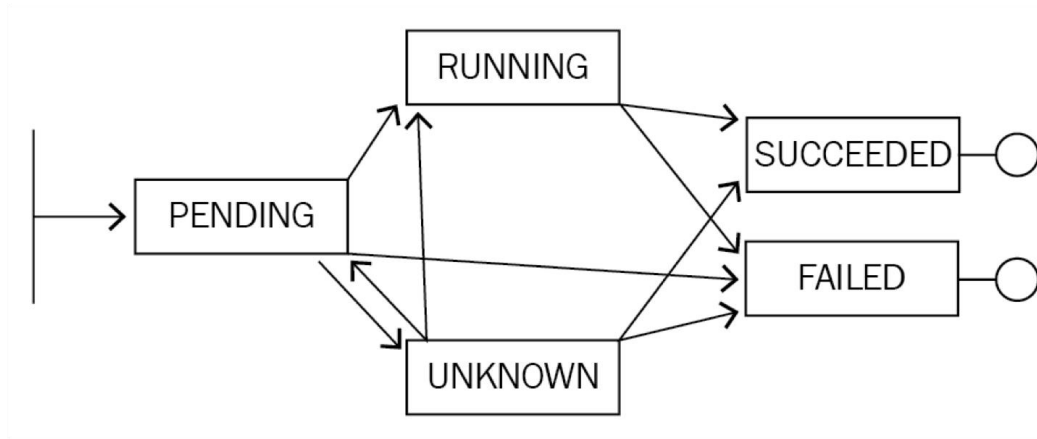
# Kubernetes生态下的监控类产品 – Prometheus(CNCF)



[https://help.aliyun.com/document\\_detail/94622.html](https://help.aliyun.com/document_detail/94622.html)



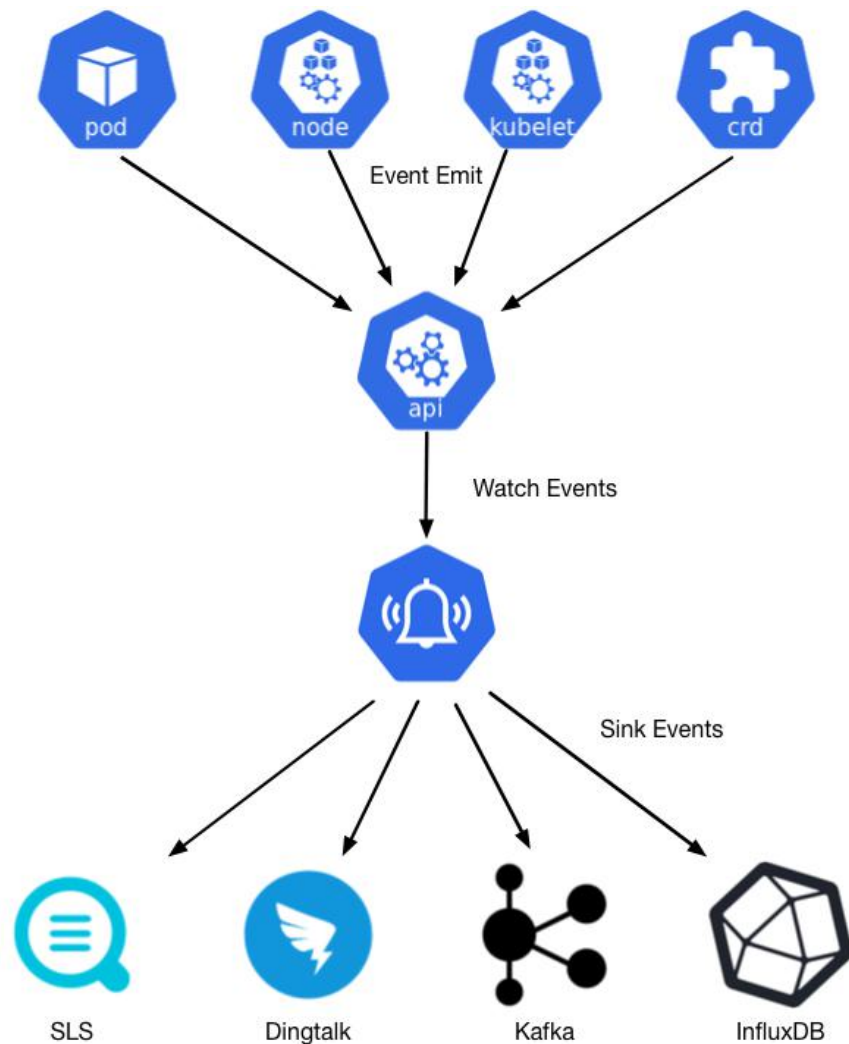
# Kubernetes生态下的监控类产品 – kube-eventer



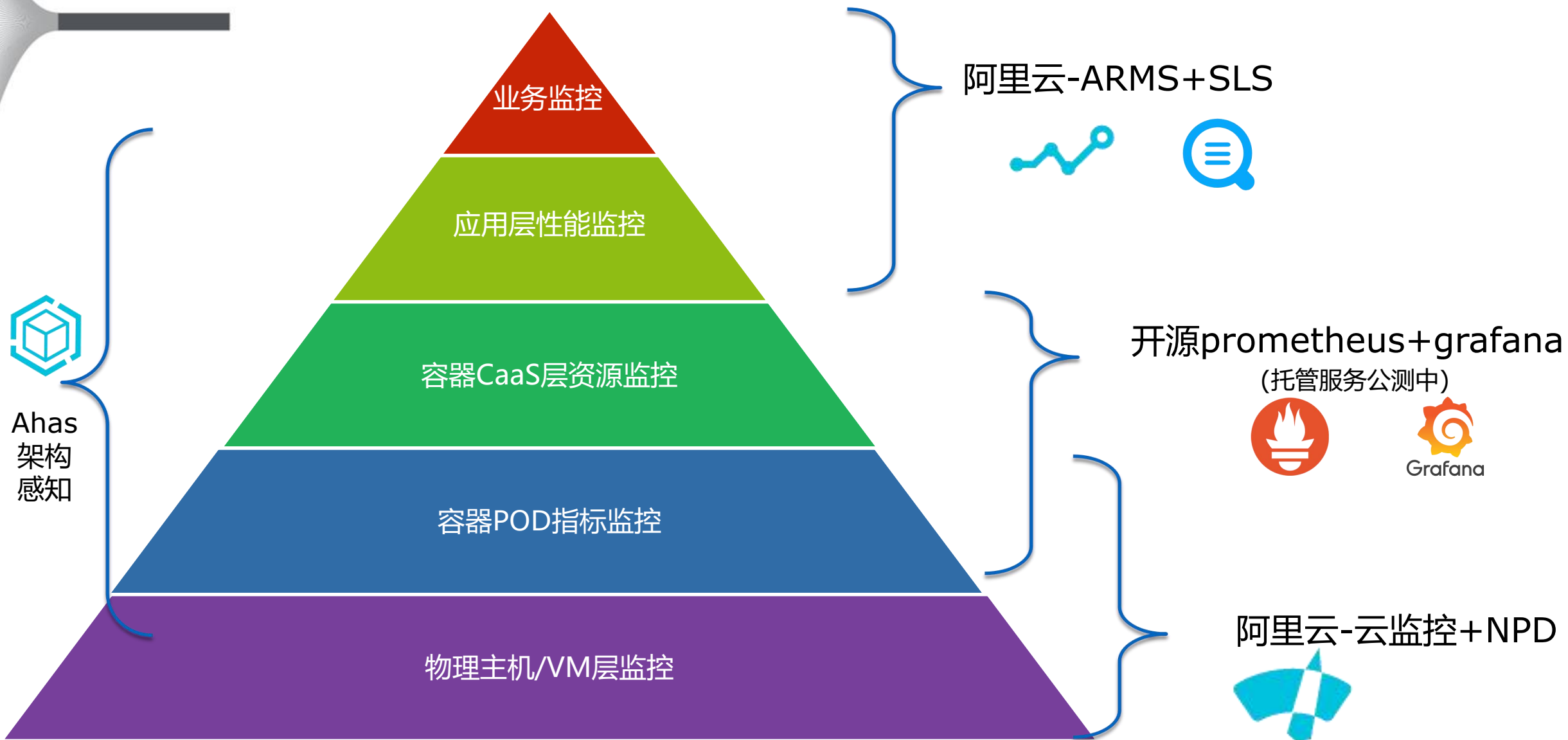
事件监控是Kubernetes中独有的一种监控方案，结合Kubernetes的状态机设计理念，可以将任何异常转换的状态告警出来。另外开发者也可以复用这条报警链路，自定义需要报警的事件，并通过离线链路报警出来。

<https://github.com/AliyunContainerService/kube-eventer>

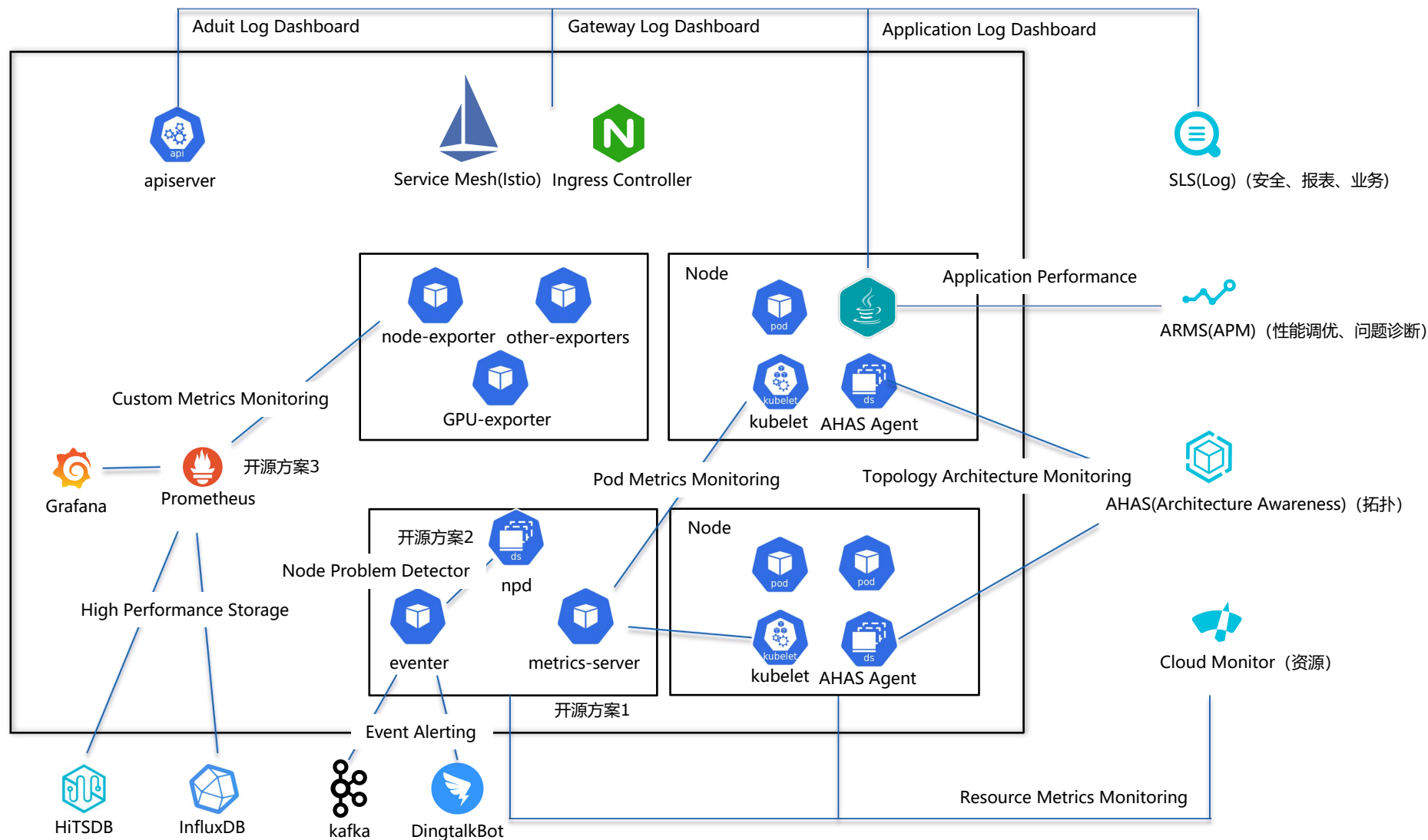
<https://github.com/ringtail/kubernetes-events-generator>



# 阿里云容器监控管理-多维度一体化监控体系



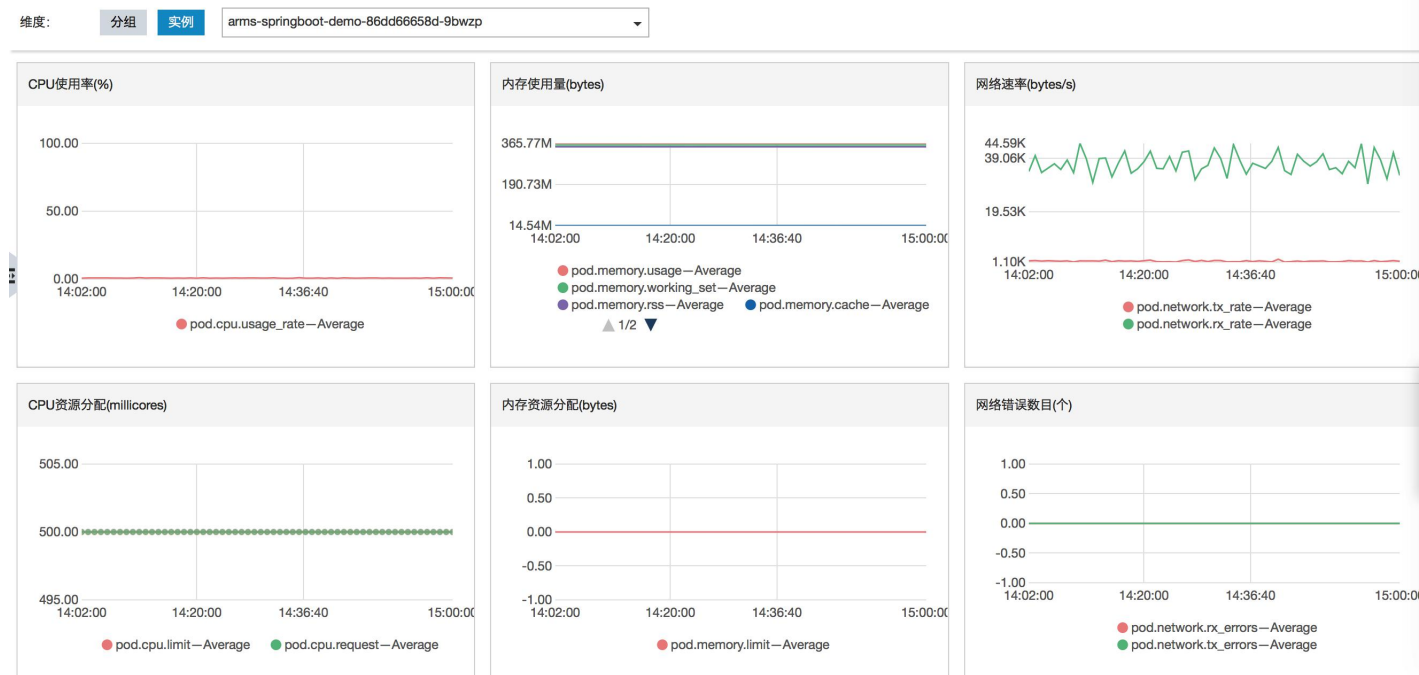
# 阿里云容器监控管理-组件与方案



# 阿里云容器监控管理-基础资源监控

## 依托阿里云监控：需先升级监控Agent

- 在K8S集群监控视图大盘
- 在K8S节点监控：每个节点/Master维度/Node维度
- Master组件/Node组件维度
- Deployment维度
- POD实例维度



# 阿里云容器监控管理-架构感知监控

自动捕获系统架构，多维度展示，从主机->容器->进程（端口）

不同层次的所属关系、依赖关系，流量关系，清清楚楚

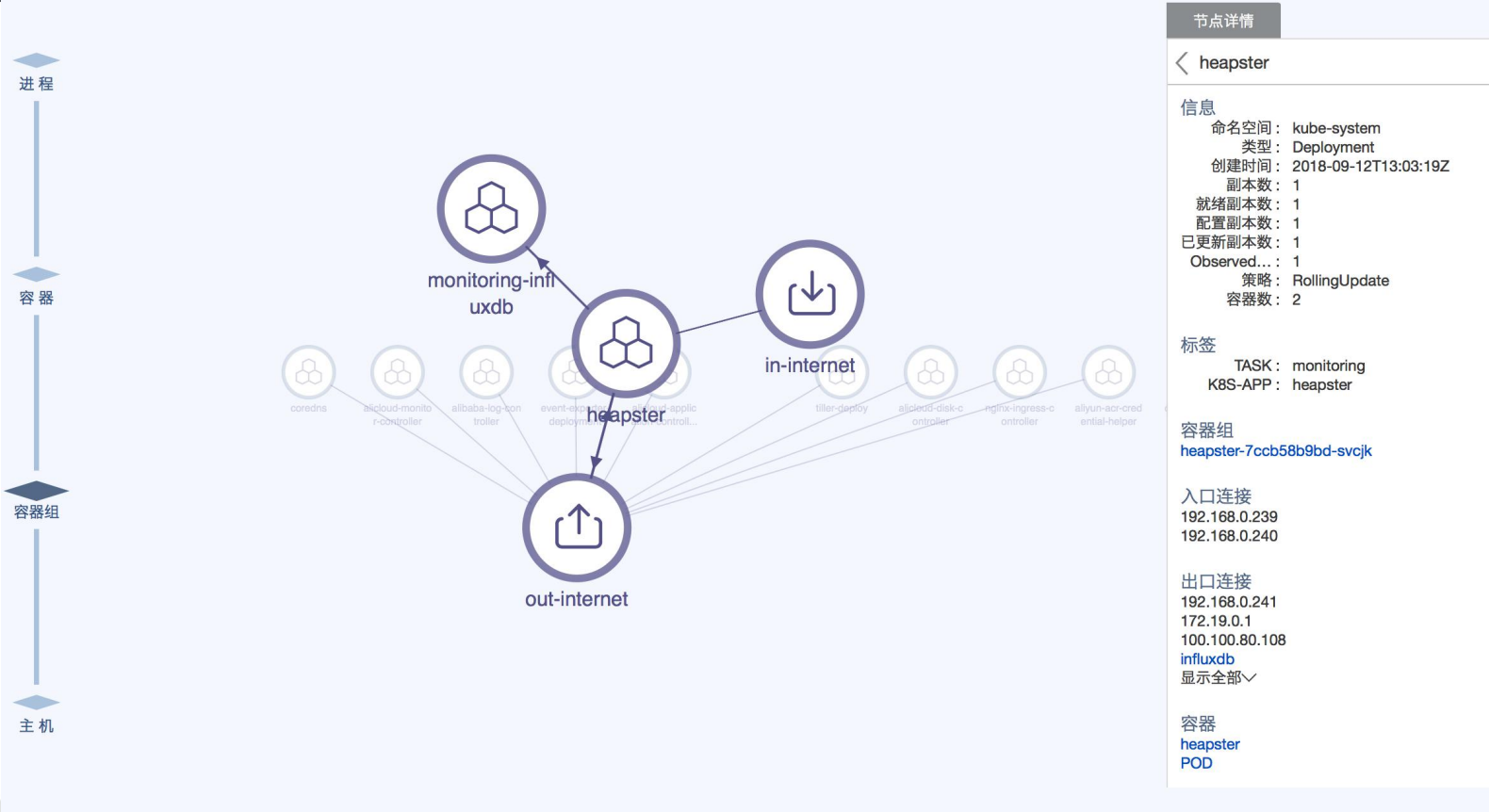
持续记录，可跟踪变化

多AZ拓扑关系(开发中)

各类云资源详情



Powered By AHAS



# 阿里云容器监控管理-应用数据盯屏与大盘

- 当前PV/UV (延迟3~5m)
- 平均延迟/P95/P99/P9999
- 成功率/500 占比
- 来源中国分布/全球分布
- 客户终端设备类型以及占比
- 当天PV/UV曲线以及昨天/上周同期曲线对比
- 来源TOP10省份/TOP10城市
- TOP10最高延迟/TOP10访问URL
- TOP10  $\geq 500$ URL/TOP10 404URL

核心数据

核心图/曲线

核心TOP10

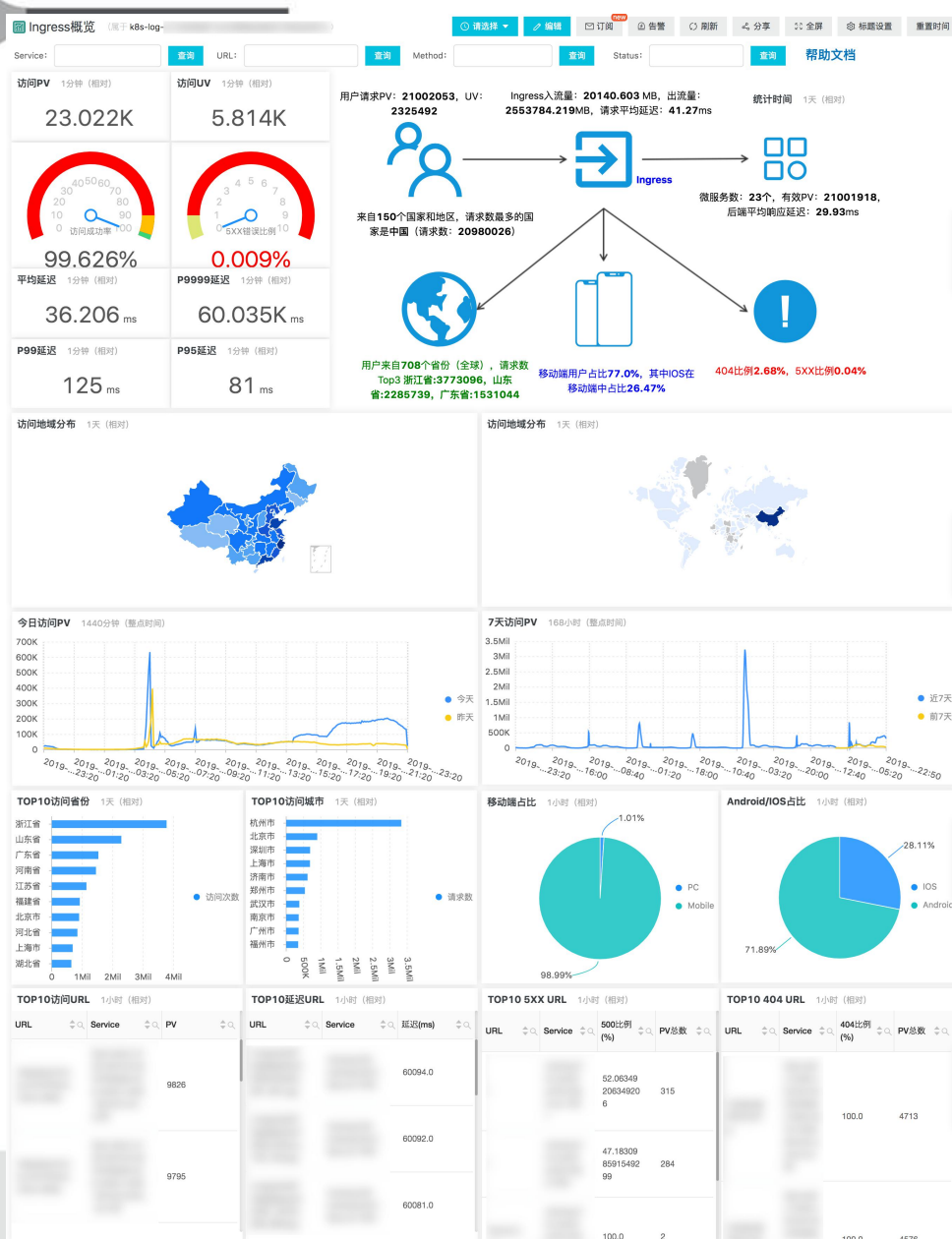


分域名分类展示、  
订阅、告警

Powered By SLS



# 阿里云容器监控管理-Ingress Dashboard展示



无需构建Es等庞大系统，系统默认提供，开箱即用

只根据存储收费， 无需维护，成本最优

涵盖业务核心场景，可扩展

逐步迭代增强，为业务保驾护航

[https://help.aliyun.com/document\\_detail/86532.html](https://help.aliyun.com/document_detail/86532.html)

# 阿里云容器监控管理-Ingress Logtail调优

1

`kubectl edit ds -n kube-system logtail-ds` # 增加环境变量和内存上限:

```
- name: cpu_usage_limit
  value: "4"
- name: mem_usage_limit
  value: "2048"
- name: max_bytes_per_sec
  value: "50000000"
- name: send_request_concurrency
  value: "100"
- name: process_thread_count
  value: "2"
resources:
  limits:
    memory: 2Gi
    cpu: "2"
  requests:
    cpu: 100m
    memory: 256Mi
```

2

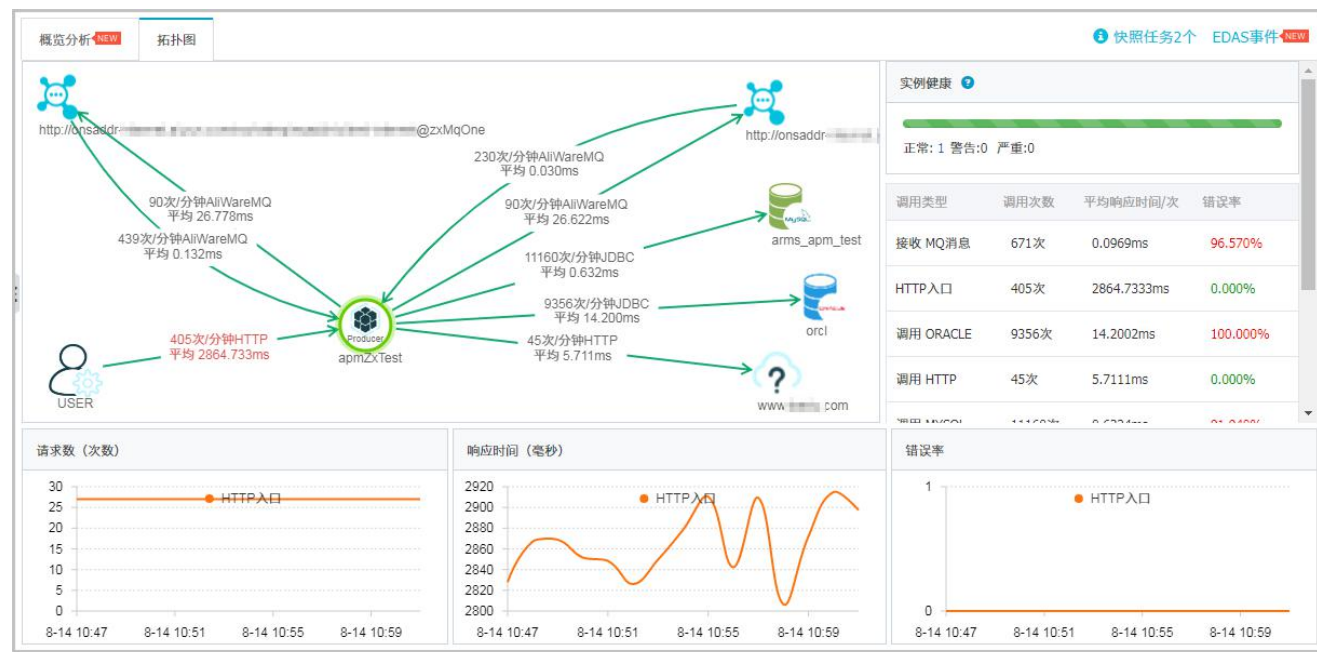
Logtail做sharding分裂

[https://help.aliyun.com/document\\_detail/48998.html](https://help.aliyun.com/document_detail/48998.html)

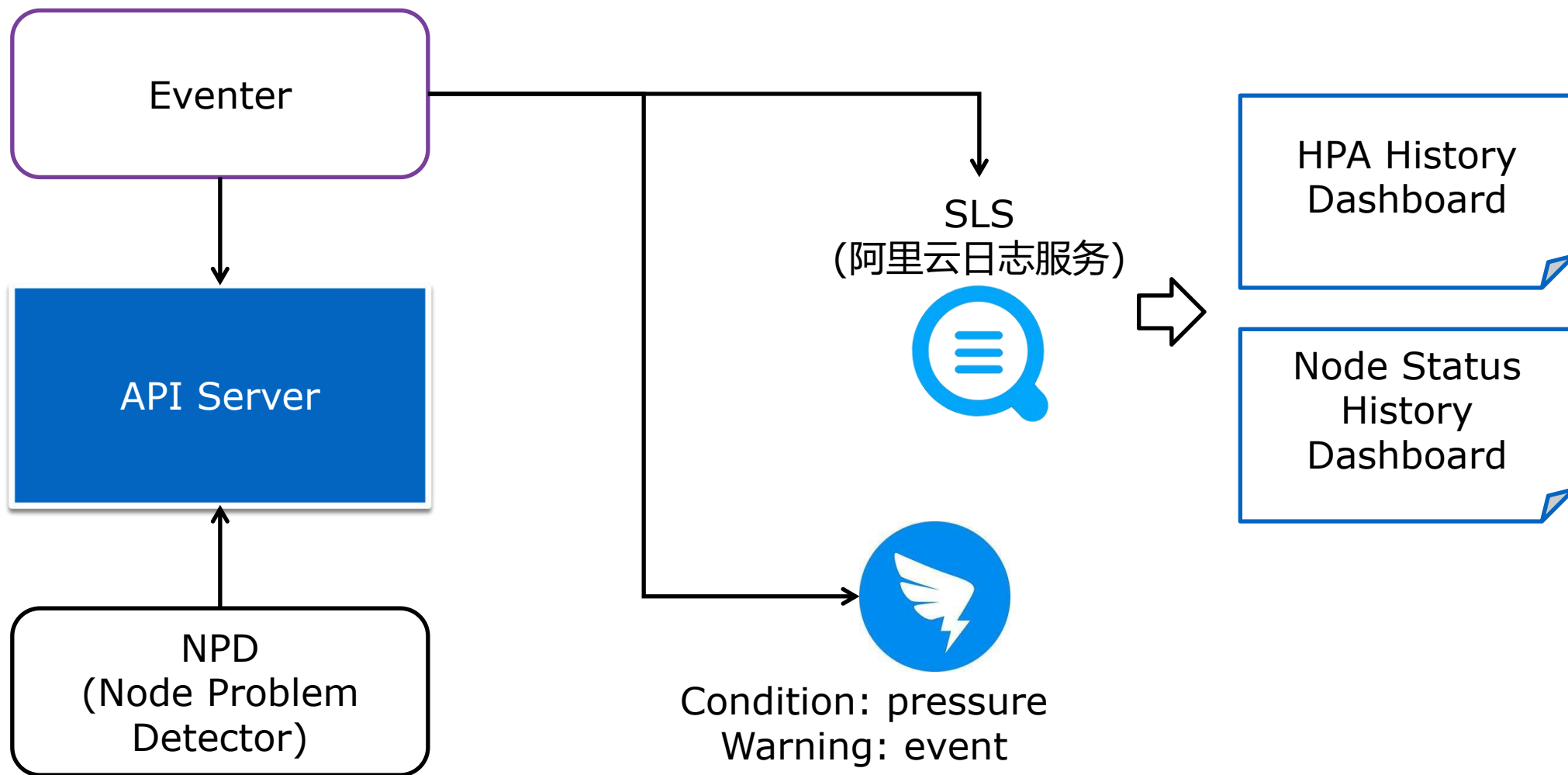
调优后可以到达 4W TPS/s (单Node, 40MB/s)

# 阿里云容器监控管理-应用性能监控

- 全局应用层拓扑
- 全局应用性能统计与分析（Dashboard）
- 调用链分析：慢调用，异常调用
- 异常快速定位与告警
- 方法栈调用分析
- JVM监控、快速定位与分析、告警
- SQL慢调用分析
- 动态生效配置，生产级适配
- 业务日志结合，调用链与业务逻辑关联分析



# 阿里云容器监控管理-事件监控可视化与报警



# 阿里云容器监控管理-事件中心

Kubernetes事件中心V1.1 (属于 kubernetes-logs)

时间选择 订阅 告警列表 刷新 分享 全屏 重置时间

事件等级: 请选择 查询

事件类型: 请选择 查询

Host: 请选择 查询

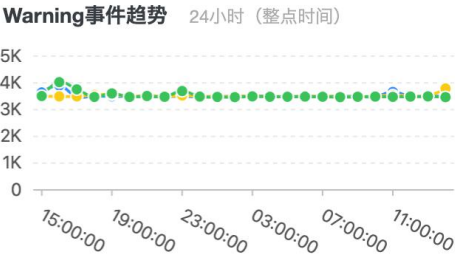
Namespace: 请选择 查询

Name: 请选择 查询



84.508K 次 ↑73  
Warning事件数/对比昨天

0  
Error事件数/对比昨天

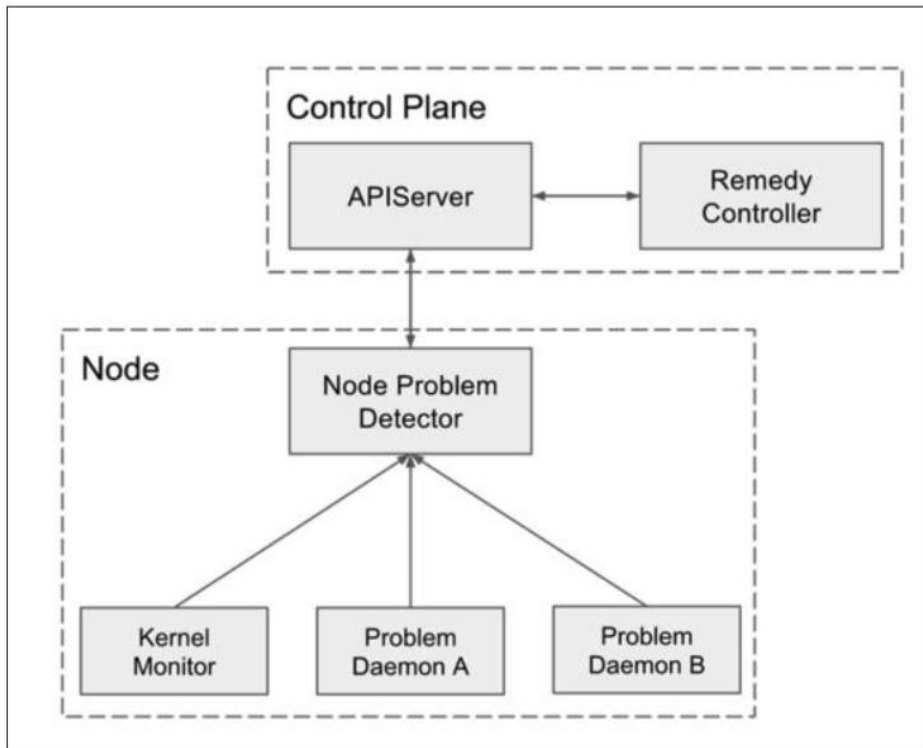


Error事件趋势 24小时 (整点时间)

暂无数据

<div>DockerHung 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>镜像拉取失败 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>驱逐 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>Pod OOM 今天 (相对)</div> <div>3 次 <span>↓-2次</span></div> <div>次数/对比昨天</div>	<div>Pod Pending 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>
<div>Pod启动失败 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>资源不足 今天 (相对)</div> <div>0 次 <span>↓-164次</span></div> <div>次数/对比昨天</div>	<div>节点重启 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>节点OOM 今天 (相对)</div> <div>0 次 <span>↓-54次</span></div> <div>次数/对比昨天</div>	<div>节点磁盘空间不足 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>
<div>节点FD不足 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>节点Pid不足 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>	<div>节点PLEG告警 今天 (相对)</div> <div>0 次</div> <div>次数/对比昨天</div>		

# 阿里云容器监控管理-利用NPD增强node错误检测



- Kernel死锁, Hung, OOMKilling
- Docker Deaon Hung住
- 虚拟网卡回收异常
- Docker文件系统异常


开源支持

```
Warning TaskHung 4m kernel-monitor, cn-beijing.i-2zecgek3bsqe4qf9anln task docker:
7 blocked for more than 300 seconds.
Normal DockerHung 4m kernel-monitor, cn-beijing.i-2zecgek3bsqe4qf9anln Node conditi
on KernelDeadlock is now: True, reason: DockerHung
Warning CorruptDockerImage 9s docker-monitor, cn-beijing.i-2zecgek3bsqe4qf9anln Error trying
v2 registry: failed to register layer: rename /var/lib/docker/image/test /var/lib/docker/image/ddd: directory not empty.*
```



# 阿里云容器监控管理-密切监控线程/进程和文件句柄泄漏

- Centos7. 4的内核3. 10还没有支持cgroup对于pid/fd限制
- Kubelet尚未支持类似docker run的--pids-limit的限制
- 阿里云增强版NPD增加主机层面的fd的监控
- K8s 1.11.1开始支持pid的condition信息

 了哥-eventer-demo 机器人

cbc5cdcd6fbb64c05982cc326a88aec2e

Level:Warning

Kind:Node

Namespace:default

Name:ap-southeast-1-i-t4n4l1c59ledtkq4306.15954f931bf45b75

Reason:TooManyOpenFiles

Timestamp:2019-04-14 10:12:19 +0000 UTC

Message:current fd is 2823 and max is 1048576 more than 80%!f(MISSING)d has been used(2823:104857

Node级别的  
钉钉定期告警(240s)

Conditions:						
Type	Status	LastHeartbeatTime	LastTransitionTime	Reason	Message	
----	-----	-----	-----	-----	-----	
FDPressure	False	Sun, 14 Apr 2019 18:13:07 +0800	Sun, 14 Apr 2019 17:41:50 +0800	NodeHasNoFDPressure	node has no fd pressure	
KernelDeadlock	False	Sun, 14 Apr 2019 18:13:07 +0800	Sun, 14 Apr 2019 17:41:49 +0800	KernelHasNoDeadlock	kernel has no deadlock	
ReadonlyFilesystem	False	Sun, 14 Apr 2019 18:13:07 +0800	Sun, 14 Apr 2019 17:41:49 +0800	FilesystemIsReadOnly	Filesystem is read-only	
NetworkUnavailable	False	Mon, 18 Mar 2019 20:53:37 +0800	Mon, 18 Mar 2019 20:53:37 +0800	RouteCreated	RouteController created a route	
OutOfDisk	False	Sun, 14 Apr 2019 18:13:26 +0800	Mon, 18 Mar 2019 23:13:27 +0800	KubeletHasSufficientDisk	kubelet has sufficient disk space available	
MemoryPressure	False	Sun, 14 Apr 2019 18:13:26 +0800	Mon, 18 Mar 2019 23:13:27 +0800	KubeletHasSufficientMemory	kubelet has sufficient memory available	
DiskPressure	False	Sun, 14 Apr 2019 18:13:26 +0800	Mon, 18 Mar 2019 23:13:27 +0800	KubeletHasNoDiskPressure	kubelet has no disk pressure	
PIDPressure	False	Sun, 14 Apr 2019 18:13:26 +0800	Mon, 18 Mar 2019 20:51:03 +0800	KubeletHasSufficientPID	kubelet has sufficient PID available	
Ready	True	Sun, 14 Apr 2019 18:13:26 +0800	Mon, 18 Mar 2019 23:13:27 +0800	KubeletReady	kubelet is posting ready status	

Addresses:

# 阿里云容器监控管理-NPD阿里云增值检查点



ntpd/chronyd没有启动



阿里云RAM Role没有配置



网络SNAT没有配置



FD数量超过水位(默认80%)



<https://github.com/AliyunContainerService/node-problem-detector>

# 阿里云容器监控管理-开源Prometheus方案



一键安装的Helm Chart（Prometheus生态组件集合）



阿里云容器服务适配的配置与模板



支持云盘/TSDB/Influxdb的集成



支持GPU独占/共享监控



集成metrics-adapter提供消费链路支持（HPA）



定制化监控大盘，支持多种场景（微服务、计算任务、机器学习）

应用目录 - ack-prometheus-operator



ack-prometheus-operator

incubator

Provides easy monitoring definitions for Kubernetes services, and deployment and management of Prometheus instances.

说明 参数

```
1 # Default values for prometheus-operator.
2 # This is a YAML-formatted file.
3 # Declare variables to be passed into your templates.
4
5 ## Provide a name in place of prometheus-operator for 'app:' labels
6 ##
7 nameOverride: ""
8
9 ## Provide a name to substitute for the full names of resources
10 ##
11 fullnameOverride: ""
12
13 ## Labels to apply to all resources
14 ##
15 commonLabels: {}
16 # scmhash: abc123
17 # myLabel: aakkmd
18
19 ## Create default rules for monitoring the cluster
20 ##
21 defaultRules:
22   create: true
```

创建

仅支持 Kubernetes 版本 1.8.4 及以上的集群。对于 1.8.1 版本的集群，您可以在集群列表中进行“集群升级”操作。不支持 Serverless Kubernetes 集群。

集群

kubeCon专用

命名空间  
monitoring

发布名称

ack-prometheus-operator

创建

# 阿里云容器日志管理

