CLOUD NATIVE + OPEN SOURCE

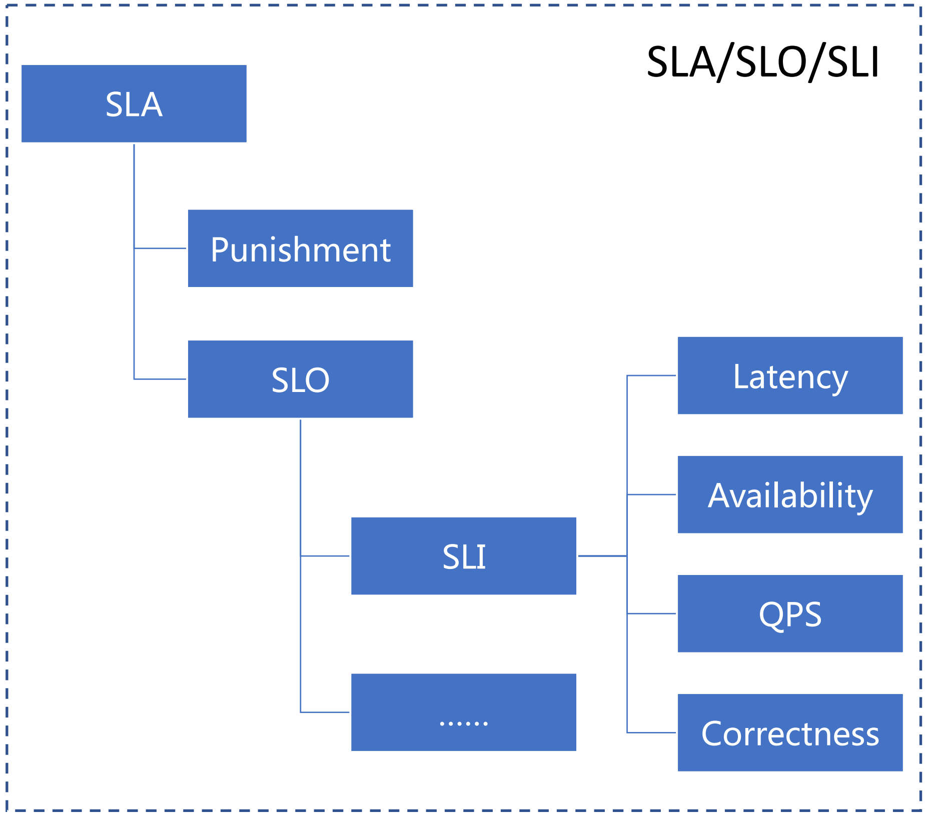
Virtual Summit China 2020

Methods to achieve high SLOs on a large scale Kubernetes cluster

Kang FAN, Jinghua YAO

Why SLO?

SLO (Service-Level Objective). Within service-level agreements (SLAs), SLOs are the objectives that must be achieved — for each service activity, function and process — to provide the best opportunity for service recipient success. — Gartner



SLI defines an indicator, which can represent user experience.

SLO is the object that try to meets all SLIs in a period of time.

SLA = SLO + Punishment.

What we concern about Large k8s Cluster

Is the cluster healthy

1 Are all software components working fine

2 How many failures occurred on the cluster

What happened about the cluster

1 Is there something unexpected happened in the cluster

2 What end users did in the cluster

How to locate failure

1 Which component is going wrong

2 Which component that leads delivery of the pod to failure

SLIs on Large k8s Cluster

1. Cluster health state

A combination value indicates the risk in the cluster. Currently Healthy, Warning and Fatal are the possible value.

2. Success rate

A rate value indicates the rate of success about creating/deleting/upgrading pod.

3. Number of Terminating Pod

A number value indicates the count of pods that can not be deleted in a certain period.

4. Centralized Components Availability

A ratio value indicates the time in which the cluster is available. It is used to evaluate the master components.

5. Nodes Availability

A number value indicates the number of unhealthy node in the cluster. Pods scheduled to unhealthy nodes may not be delivered in time, success rate would decrease consequently.

The success standard and reason classification

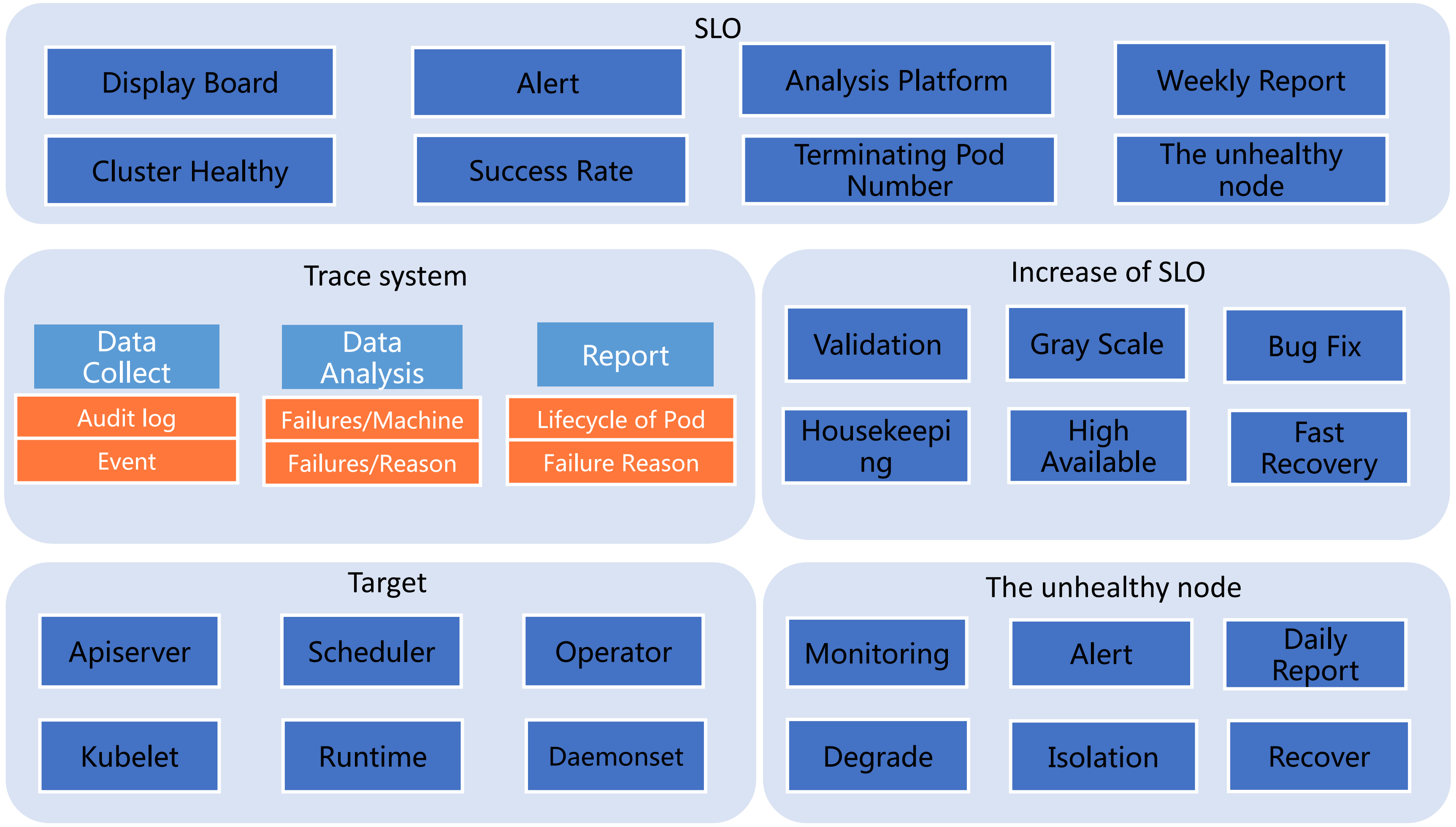
The success standard:

Pod	Feature	Time limit	Success condition
Pod	RestartPolicy=Always	1min (example value)	the status of {.Status.Conditions. “type==Ready”} is "True”
Job Pod	RestartPolicy=Never/OnFailure	1min	{.Status.Phase} == Running/Succeeded/Failed
Terminating Pod		1min	pod is removed from etcd
Unhealthy Node	Taint/Degrade	1min	Node has taints or is degraded
	Processing	Base on the failure reason	Unhealth node is healed or removed.

Reason classification:

Source	Feature	Example
System	Failure caused by cluster itself	RuntimeError, ImageFailed, Unscheduled, KubeletDelay...
End Users	Failure caused by end users	ContainerCrashLoopBackOff, FailedPostStartHook, Unhealthy...

The infrastructure



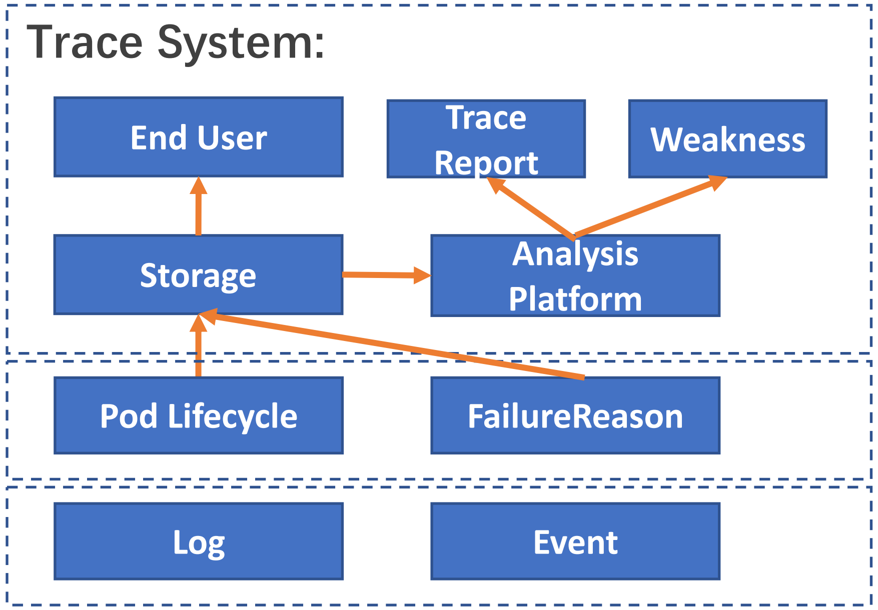
SLO ：
Indicate the cluster is healthy or there is something unexpected happened.

Trace system ：
Collect and analyze logs in cluster. So we can known what happened about the cluster.

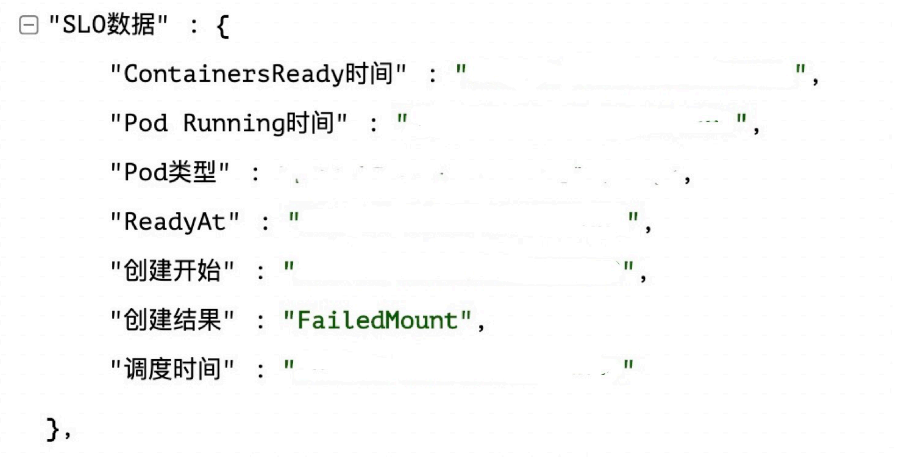
Increase of SLO ：
Get the weakness of the cluster by analyzing the failure reasons and effective actions should be taken to increase the success rate.

The unhealthy node ：
Detect the unhealthy machines timely and fix problems automatically。

The trace system



Trace Result:



```
⊟ "SLO数据" : {
      "ContainersReady时间" : "[redacted]",
      "Pod Running时间" : "[redacted]",
      "Pod类型" : [redacted],
      "ReadyAt" : "[redacted]",
      "创建开始" : "[redacted]",
      "创建结果" : "FailedMount",
      "调度时间" : "[redacted]"
  },
```

Data Collect:
Collect Audit log for the whole cluster.

Data analysis:
Analyze failure reason if pod is failed.

Reason analysis:
Analyze the failure reasons. Try to find something abnormal in the cluster.

We can get:
It is failed to deliver the pod， and the fail reason is FailedMount.

CLOUD NATIVE + OPEN SOURCE
Virtual Summit China 2020

Node Metrics

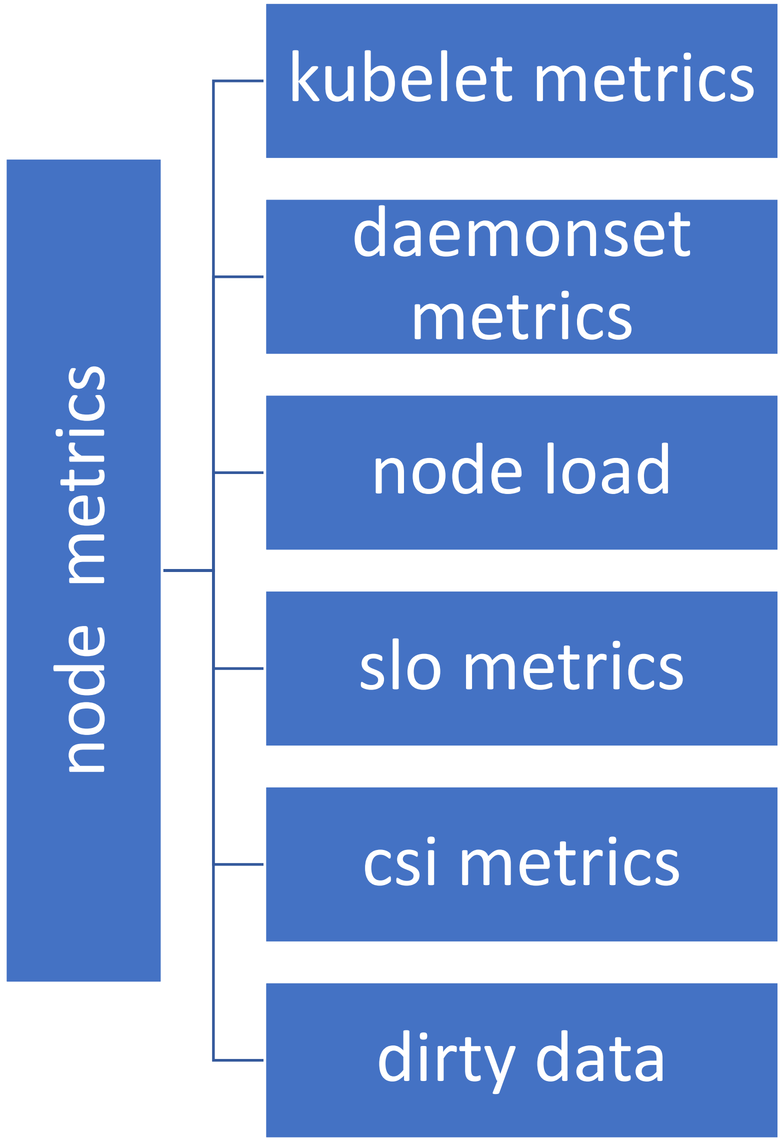
With huge amount of metrics data collected, statistical methods can be used to check whether the node is healthy or not.

Besides, node delivery capacity can also be evaluated via historical data.

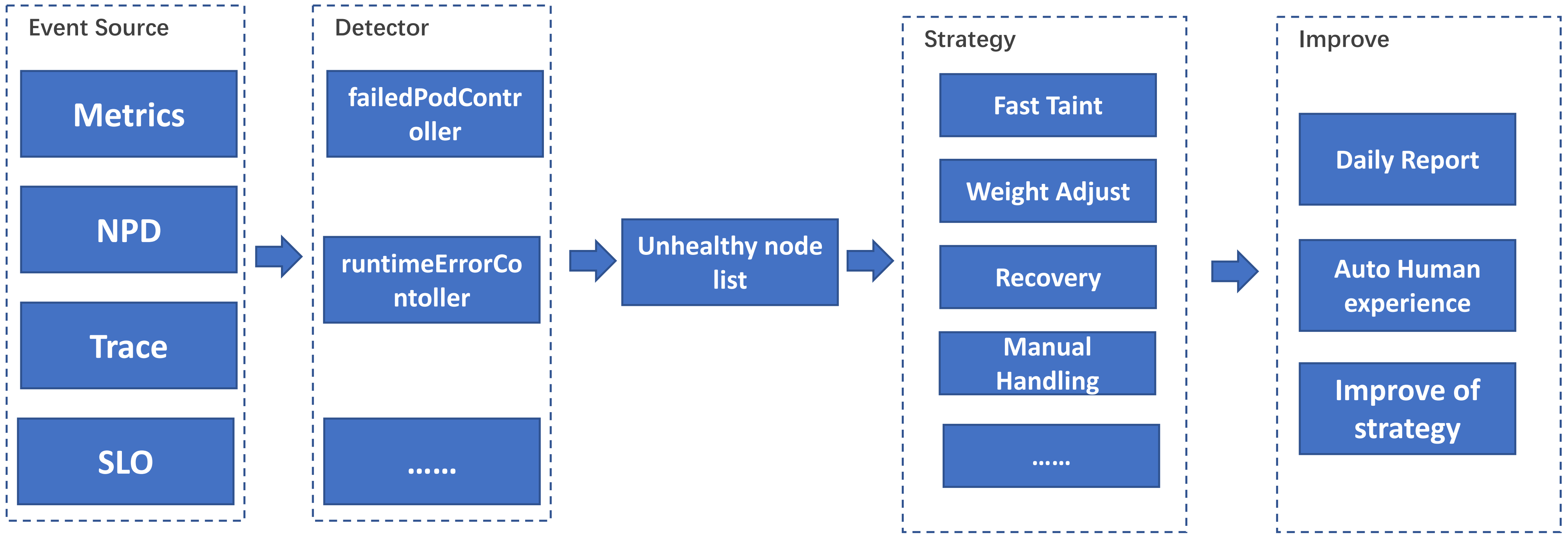
With dirty data metrics which consists of

- escaped/zombie/uninterruptible process
- orphaned containers
- orphaned pod directories/volumes
- orphaned cgroups
- orphaned net device

and so on, node recovery system can cleanup those dirty data or alert cluster admins to process dirty data manually.



Unhealthy node



1. Collect data from metrics NPD, Trace system and Log.
2. Analyze the problem of the node, such as DiskRO, critical Daemonset is not ready.
3. Processing unhealthy node: **Heal**, **Degrade** or **Isolate**. With scoring mechanism and historical operation records, unhealthy nodes can be recovered automatically. Otherwise manual intervention is required.
4. Generate daily report to show what happens, and programmers can improve the system with the reports continuously.

Tips on increasing SLO

Case 1: Image Download
Image lazyload technology provides the ability to run a container without downloading image.

Case 2: Retry
Pod should be recreate when the previous pod is failed. The previous node should be excluded.

Case 3: Critical Deamonset
Node should be tainted when critical Daemonset is unhealthy.

Case 4: Plugin registry
Registration of plugin such as CSI plugin should be checked.

Case 5: Capacity
The QPS Limit and Capacity Limit should be applied.

CLOUD NATIVE + OPEN SOURCE

Virtual Summit China 2020