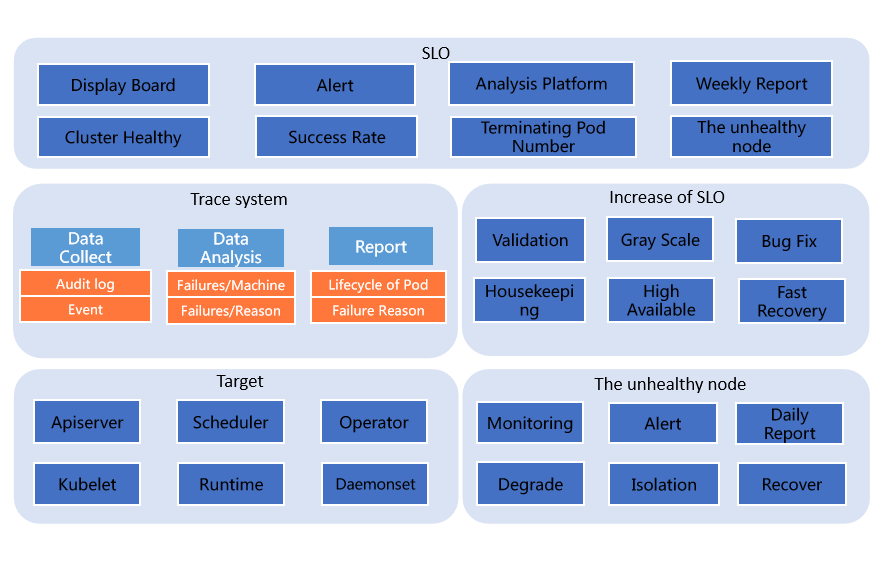
蚂蚁集团的 SLO 体系

确定好 SLO 各项关键指标的定义之后，接下来就是构建 SLO 体系。

据范康介绍，蚂蚁集团 SLO 系统主要包括两个方面，一个方面用于向终端用户 / 运维人员展示当前集群各项指标状，另一方面是各个组件相互协作，分析当前集群状态，获取影响 SLO 的各项因素，为提升集群 pod 交付成功率提供数据支持。



蚂蚁集团 SLO 体系架构图

自顶向下而看，蚂蚁集团 SLO 的分层架构包括 SLO、Trace system、Increase of SLO、Target 和 The unhealthy node。

其中，顶层组件主要面向各种指标数据， 如集群健康状态、pod 创建、删除、升级成功率、残留 pods 数量，不健康节点数量等指标。其中 Display Board 是指监控大盘，可能不会实时查看，为避免错过处理紧急事件的最佳时机，同时构建了 Alert 告警子系统，支持配置多种告警方式；Analysis System 通过分析指标历史数据以及采集到的节点 metrics 和 master 组件指标，给出更详细的集群运营报告；Weekly Report 子系统给出当前集群本周 pod 创建 / 删除 / 升级的数据统计，以及失败案例原因汇总；Terminating Pods Number 给出一段时间内集群内新增的无法通过 Kubernetes 机制删除的 Pods 列表和 Pods 残留原因；Unhealthy Nodes 则给出一个周期内集群所有节点的总可用时间占比，每个节点的可用时间、运维记录、以及不能自动恢复，需要人工介入恢复的节点列表。

为了支撑上述这些功能，蚂蚁 集团 还开发了 Trace System，用来分析展示单个 pod 创建 / 删除 / 升级失败的具体原因。其中包含日志和事件采集、数据分析、pod 生命周期展示三个模块。日志和事件采集模块采集各 master 组件以及节点组件的运行日志和 pod、node 事件，分别以 pod/node 为索引存储日志和事件；数据分析模块分析还原出 pod 生命周期中各阶段用时，判断 pod 失败原因，节点不可用原因。最后，由 Report 模块向终端用户暴露接口和 UI，向终端用户展示 pod 生命周期以及出错原因。

经验总结

目前蚂蚁集团 的 SLO 实践不仅提高了集群 pod 的交付成功率，同时通过构建 tracing 系统，分析到集群内 pod 交付关键链路的耗时，整理失败原因，实现了数据分析 / 诊断平台。对于如何实现高 SLO，范康也给出了自己的五点经验。

* 在提升成功率的进程中， SLO 治理 团队面临最大的问题是**镜像下载**。Pod 必须在规定时间内交付，而镜像下载通常需要非常多的时间。所以， 团队 通过计算镜像下载时间，专门设置了一个 ImagePullCostTime 的错误，即镜像下载时间太长，导致 Pod 无法按时交付。另外，阿里镜像分发平台蜻蜓支持了 Image lazyload 技术，在 Kubelet 创建容器时，不用再下载镜像，大大加速了 Pod 的交付速度。
* **提升单个 Pod 成功率**：随着成功率的提升，再提升的难度会越来越大，这是可以引入 workload 进行重试。蚂蚁 集团 内部的 PaaS 平台会不断重试，直到 Pod 成功交付或者超时。需要注意的是，重试时要先排除之前的失败节点。
* **检查关键 Daemonset**：如果关键 Daemonset 缺失，把 Pod 调度上去是很容易出问题的，甚至影响到创建 / 删除链路，这样可能就接入故障机体系。
* 很多 Plugin 是需要向 Kubelet 注册的，如 CNI Plugin，可能存在节点上一切正常，但向 Kubelet 注册时失败的情况，那么这个节点同样无法提供 Pod 交付的服务，需要接入故障机体系。
* 由于集群中的用户数量非常多，所以**隔离**很重要。在权限隔离的基础上，还需要做到 QPS 隔离、容量隔离，防止一个用户的 Pod 把集群能力耗尽，影响其他用户的利益。