

WOTA

51CTO

World Of Tech 2017

全球架构与运维技术峰会

2017年4月14日-15日 北京富力万丽酒店

ARCHITECTURE



出品人及主持人：

来 炜 滴滴出行基础平台部
技术总监

创新运维探索

基于Mesos/Docker构建企业级SaaS应用 —— Elasticsearch as a Service

马文 去哪儿网opsdev



马文

去哪儿网

平台事业部数据平台研发工程师

分享主题：

基于Mesos/Docker构建

Elasticsearch as a Service

主要内容：

- 背景/设计目标
- 技术调研/选型
- 细节实现
- 集群信息汇总与自助配置
- 自动化部署
- 监控与报警
- 日志/marathon事件收集
- 问题&解决

背景/设计目标

Elasticsearch需求量暴增带来的交付与维护弊端：

- 传统部署以kvm虚拟机或实体机为ES节点，需要提前创建虚拟机或申请实体机
- ES运行环境需使用ansible, salt等工具事先部署
- 节点扩容需人力一遍一遍去部署环境
- 集群不易于管理，维护成本高

背景/设计目标

针对上述弊端，我们制定的几点设计目标：

- 加快集群构建速度
- 快速扩容和快速迁移能力
- ES使用/运维标准化
- 集中的信息展示与良好的用户交互界面

背景/设计目标

容器化的私有云平台
Elasticsearch as a service (ESAAS)

技术调研

- Elastic Cloud
 - Elastic官方的公有云平台，能够做到ES服务即启即用，横向扩容，自助配置等
- Amazon Elasticsearch Service
 - 基于亚马逊AWS公有云服务的搜索和分析平台，同Elastic cloud一样，能够提供即启即用的ES服务
- Elasticsearch Framework on Mesos
 - 一个开源的基于Mesos的任务调度框架，每一个executor即为一个ES节点实例，每一个框架即为一个ES集群。功能少，无法配置多角色的节点，无法提供插件，script等自定义配置等

预期功能点

- 集群统一管理/集中的信息交付
- 集群资源quota限定/集群隔离
- 数据持久化存储/可靠性保障
- 数据节点快速水平/垂直扩容
- 集群/外围插件/script的自助配置
- 集群的发布和配置管理
- 完整的集群/平台监控和报警

技术选型



docker

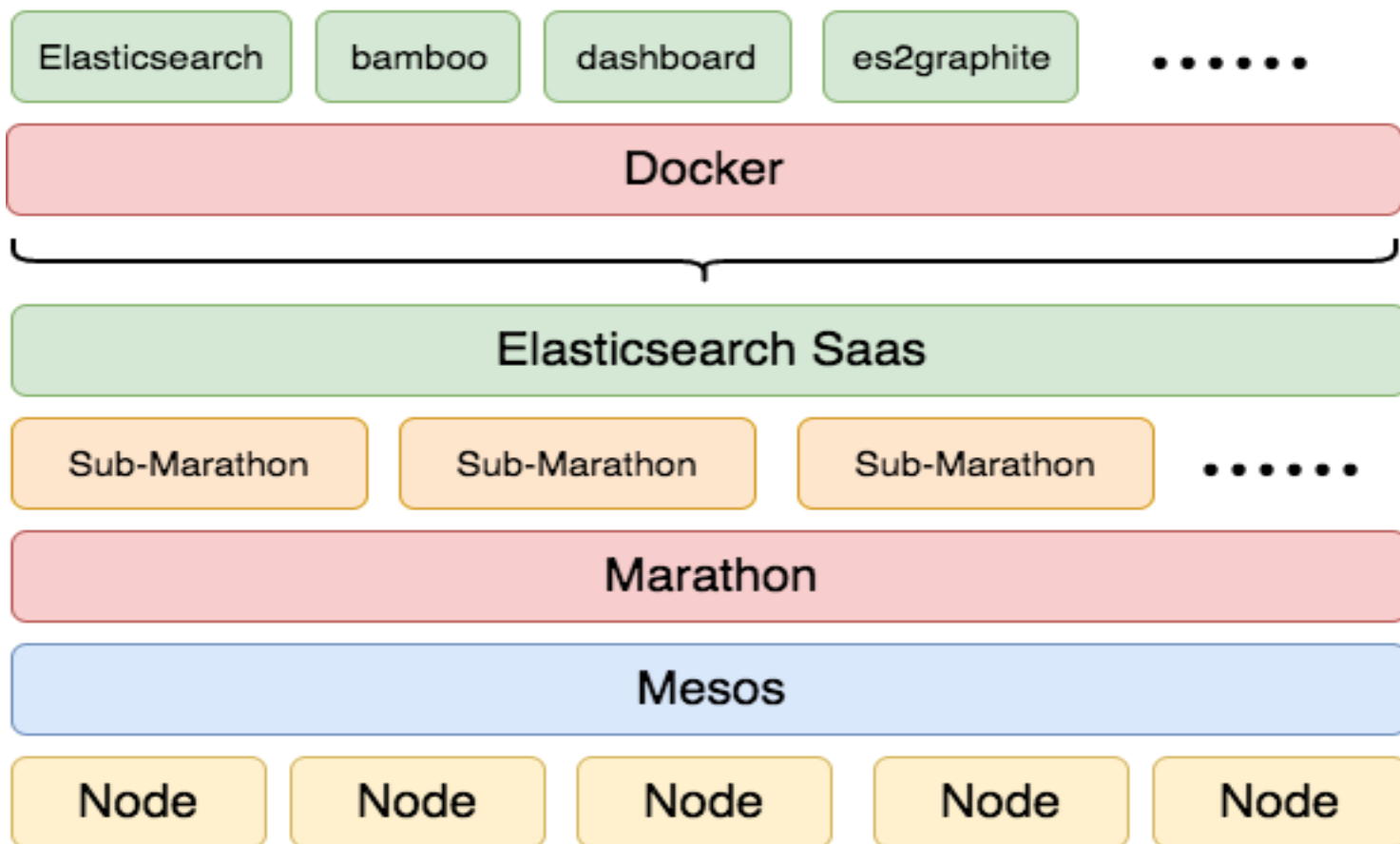


MARATHON



MESOS

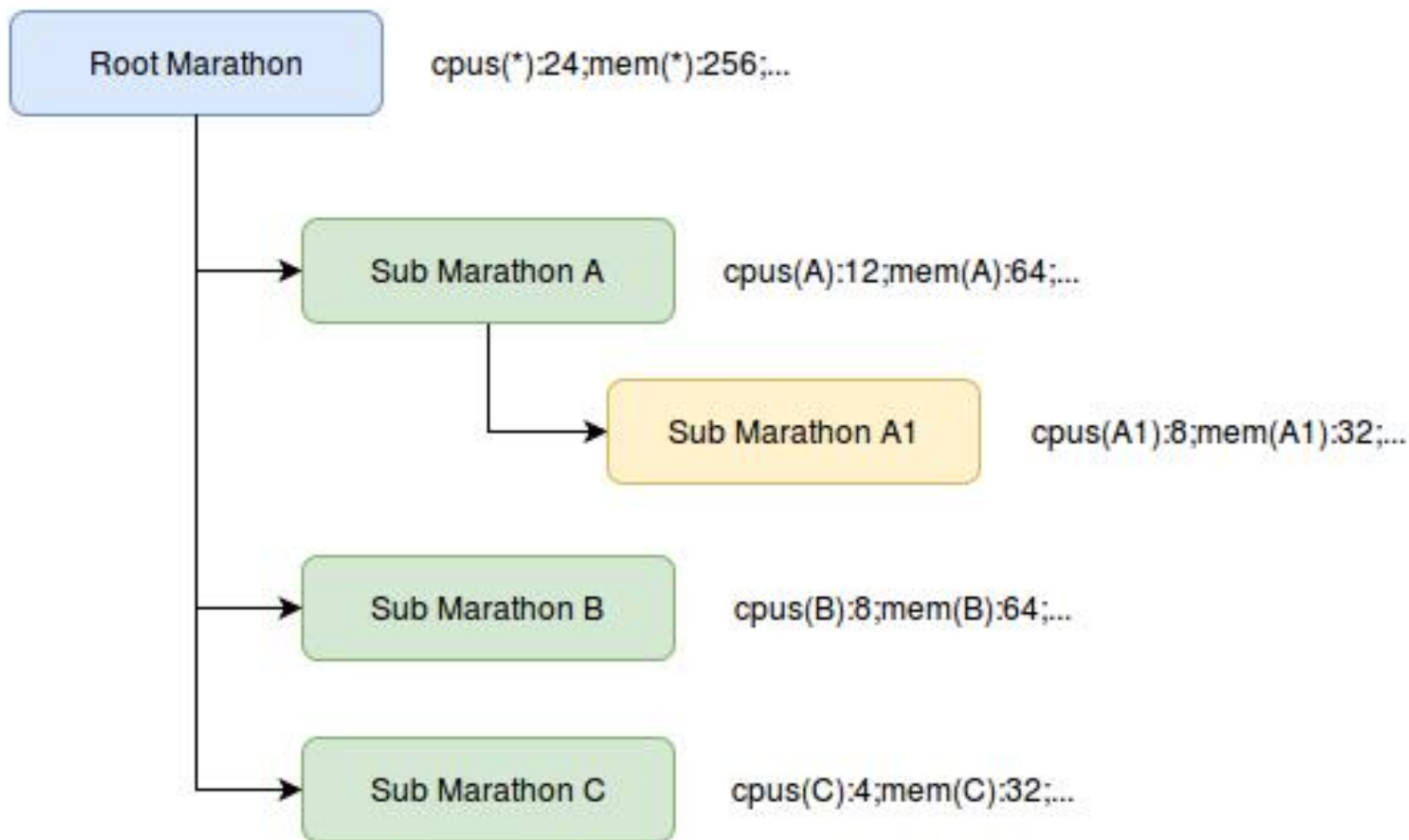
总体结构



几个核心问题

- Quota分配
- 集群的隔离
- 服务发现
- 数据可靠性
- 自助化集群配置
- 集中的交付信息展示
- 部署自动化
- 监控与报警

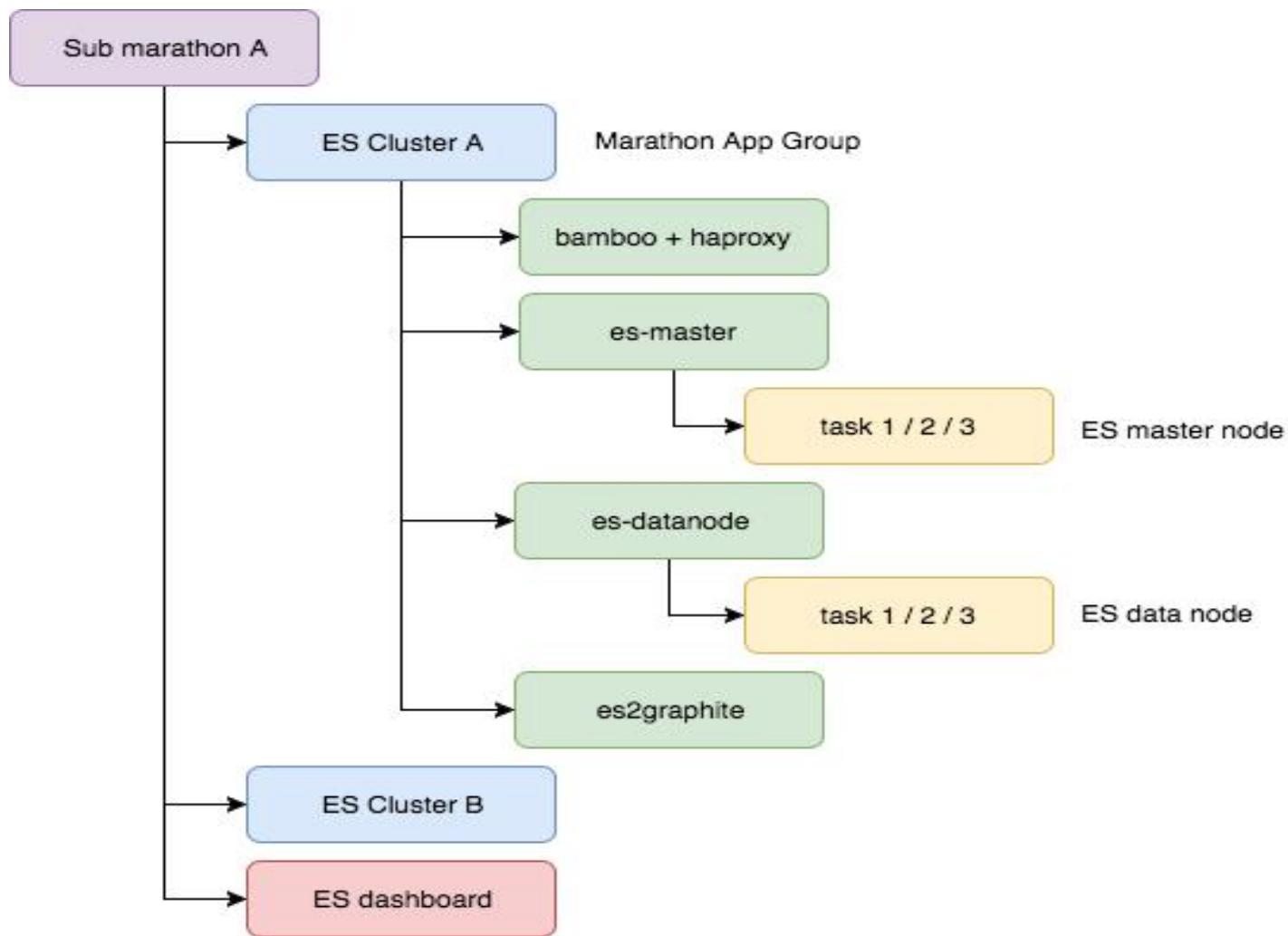
资源分配结构



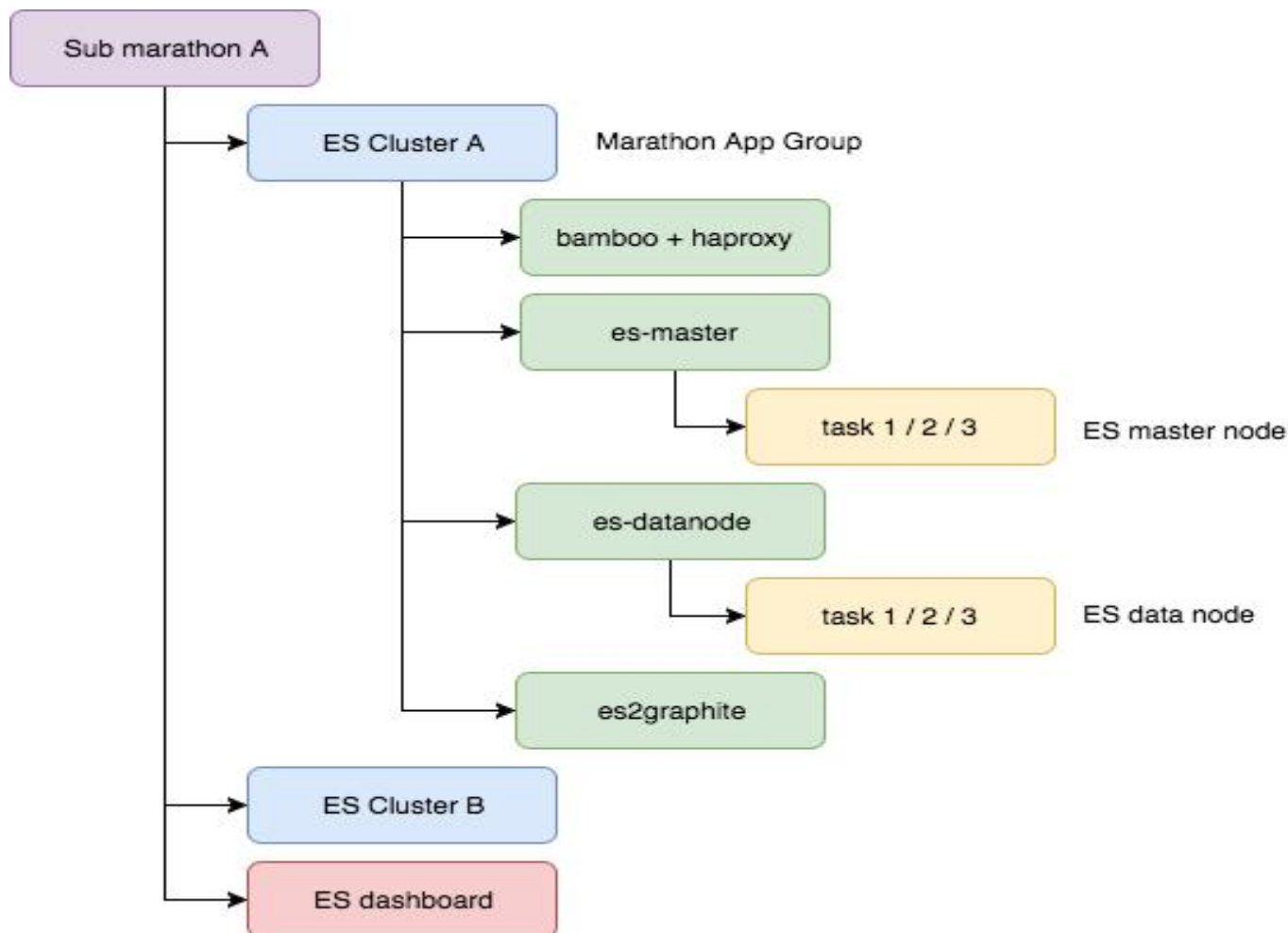
Quota & 资源划定

- Mesos Role & 动态Quota设定
- Root marathon 不做资源设定， 拥有系统全部的资源
- 每一个Sub-marathon都拥有不同的Role
- 自由Sub-marathon享有限定范围内的资源， 并具有逻辑上隔离的命名空间

集群逻辑隔离/划分

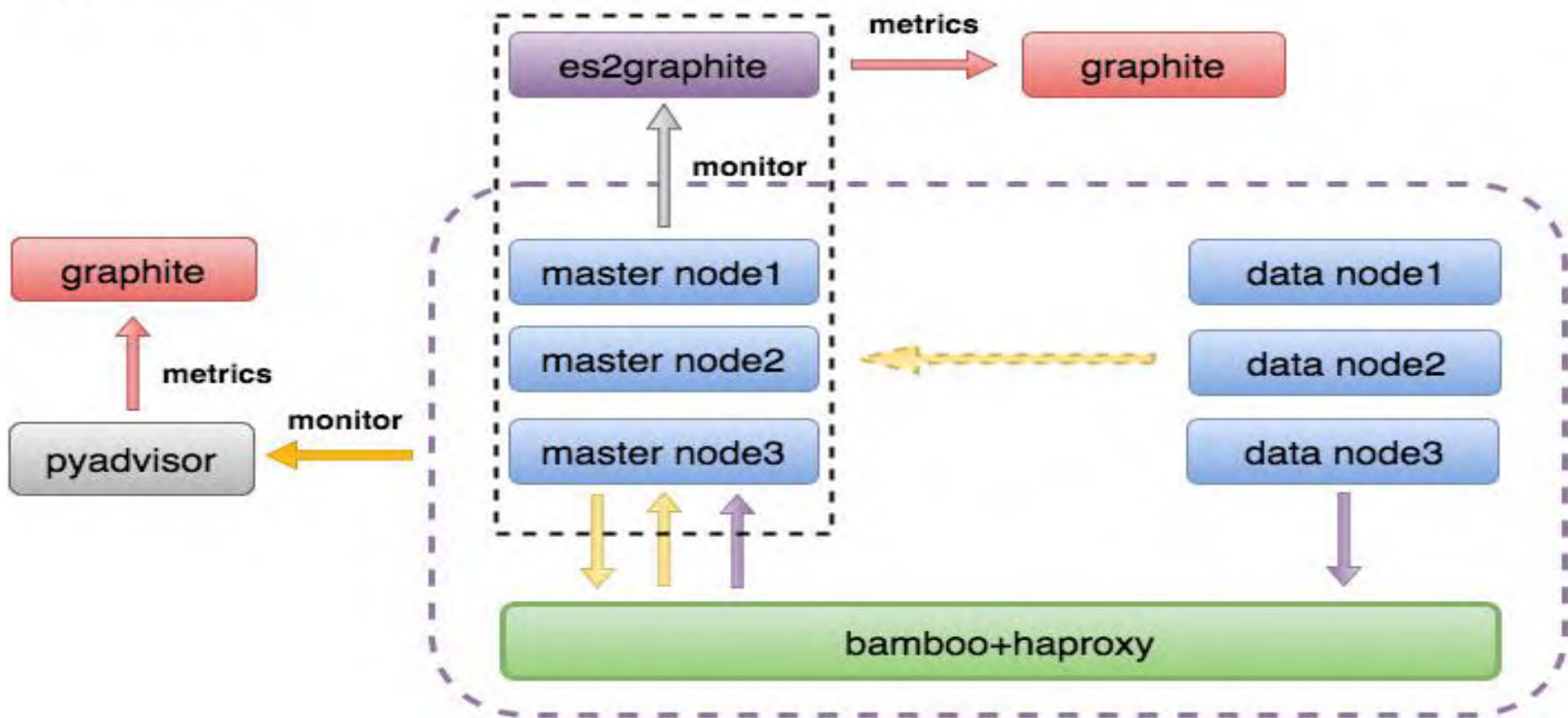


集群逻辑隔离/划分

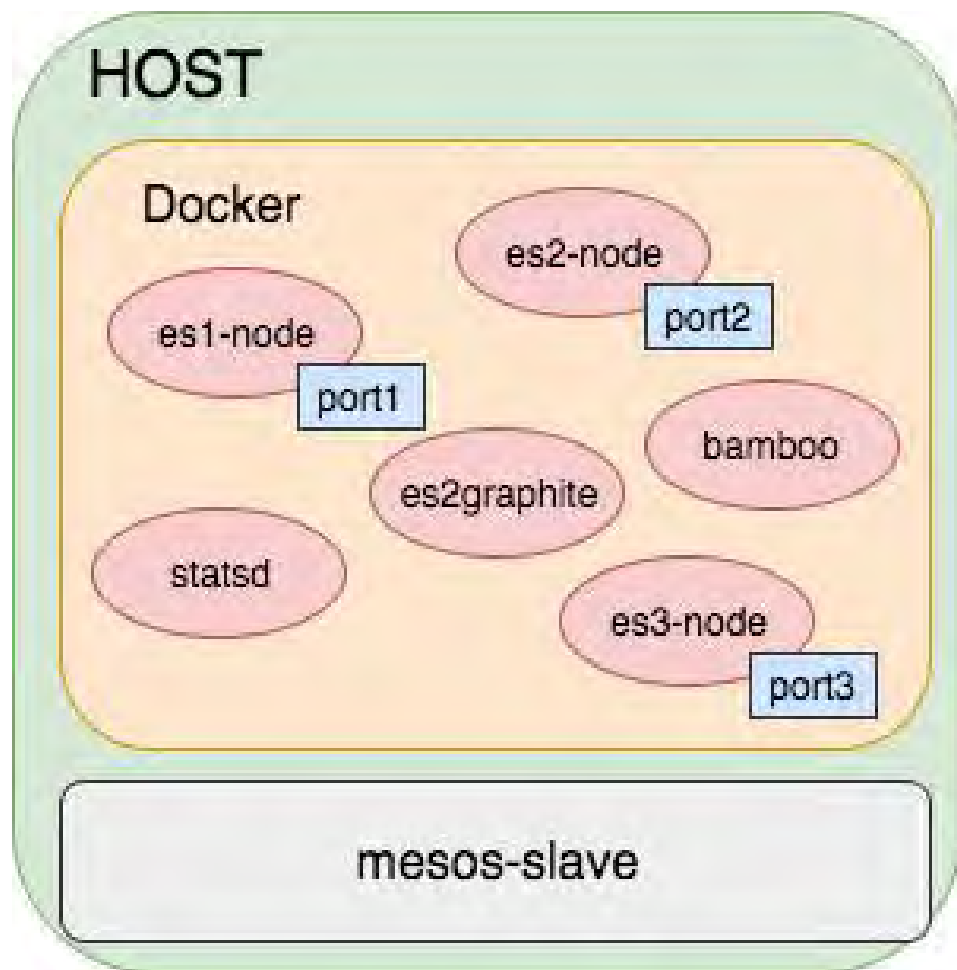


一个完整的ES集群逻辑结构

Sub-Marathon



单台物理机的结构



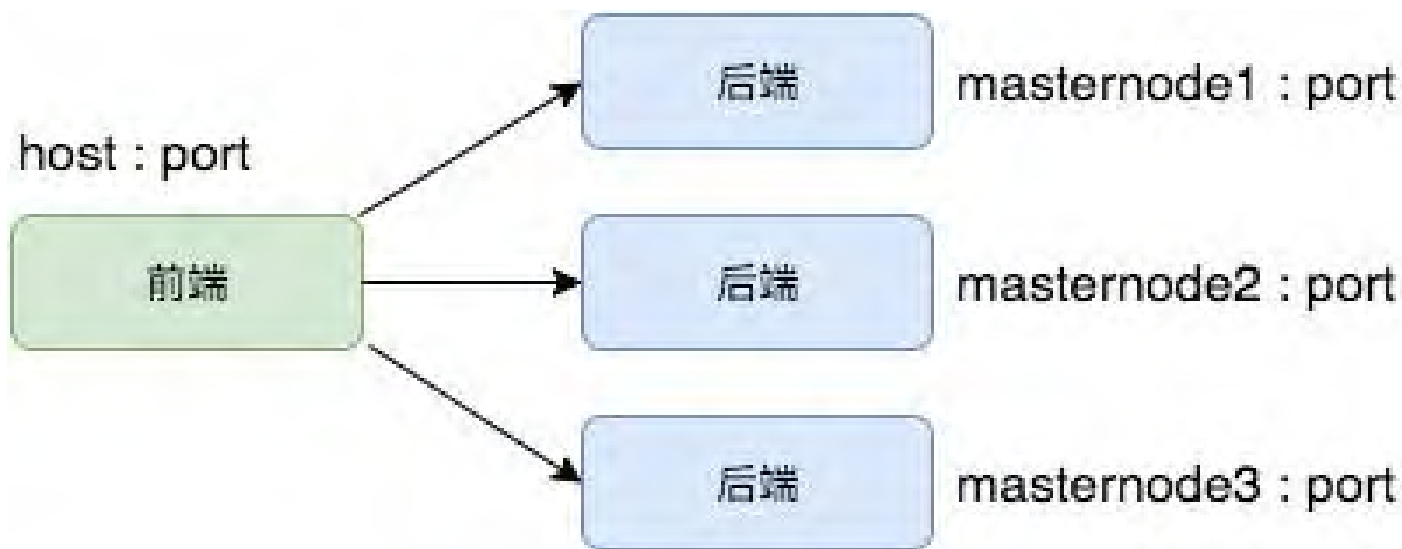
- 一个机器上可有多个ES节点
- 使用不同的端口来区别ES集群

服务发现

使用bamboo + haproxy来实现ES集群内部节点发现:

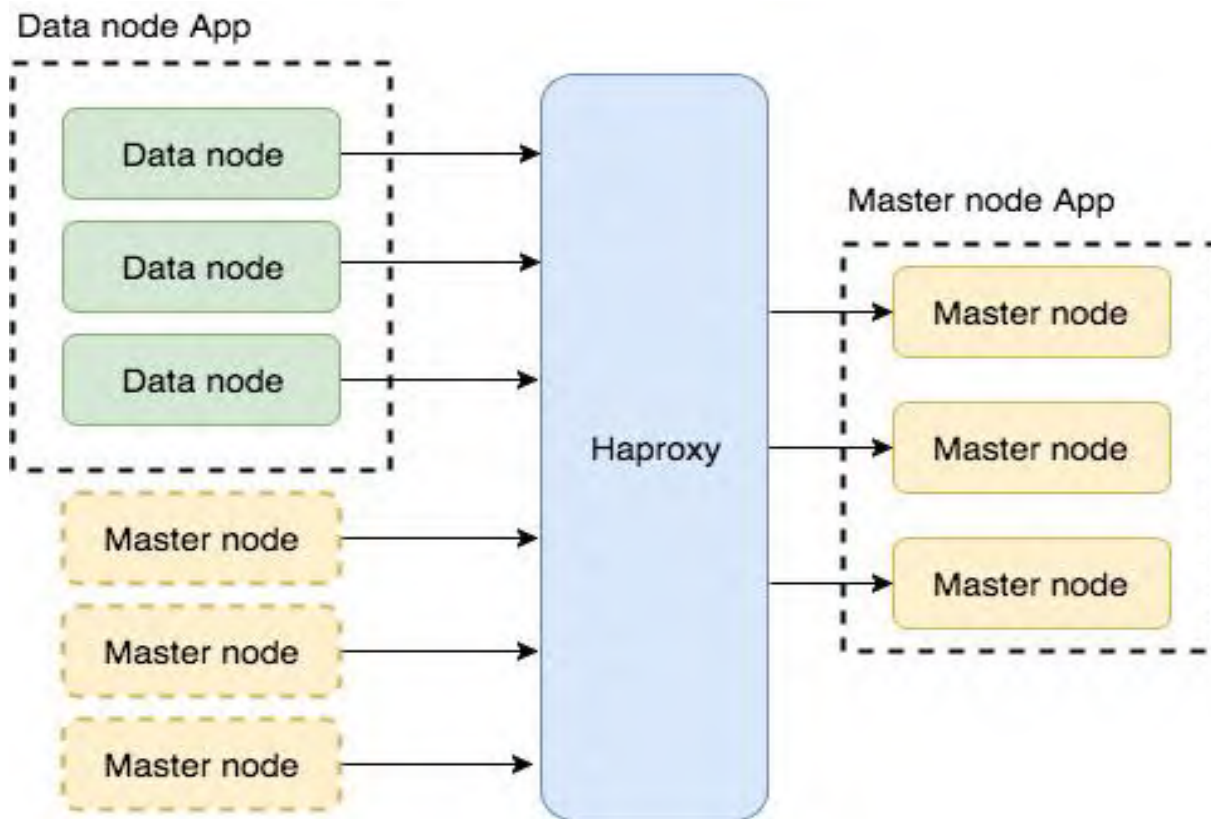
- bamboo 是一个开源的marathon服务发现工具, 通过注册marathon callback来监听marathon事件, 根据配置自动reload haproxy

bamboo + haproxy 内部端口转发示意图:



如何解决服务发现

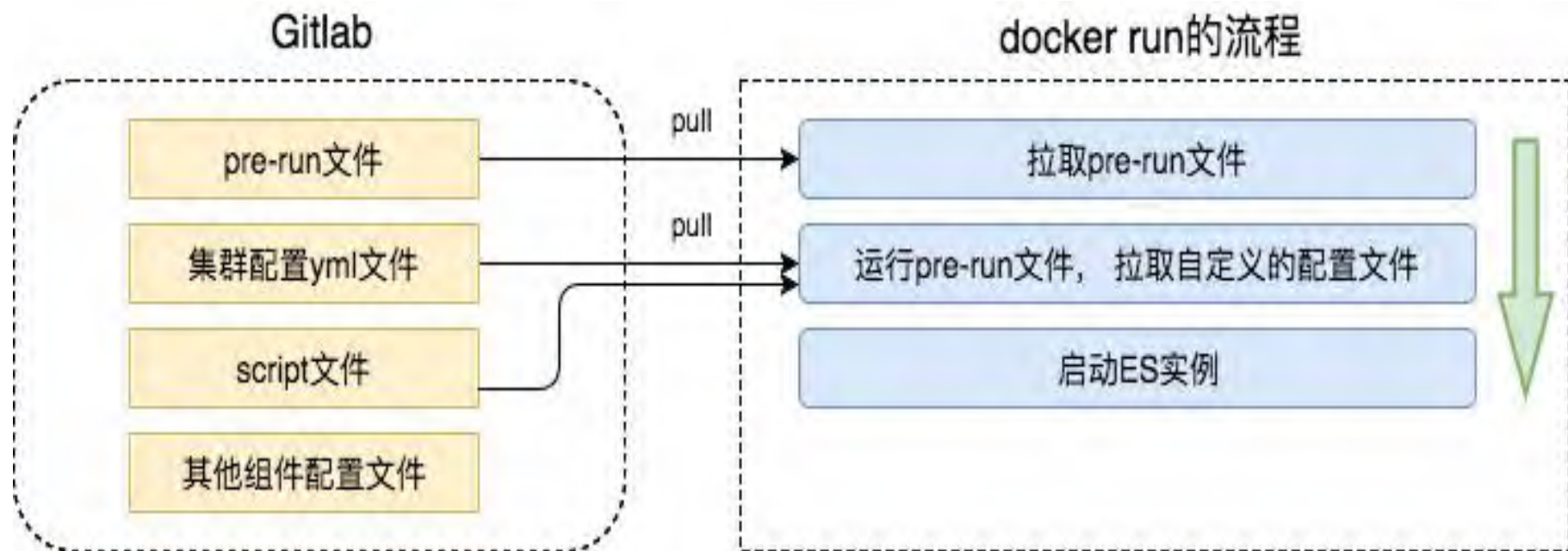
discovery.zen.ping.unicast.hosts: ["<haproxy_ip>:<haproxy:port>"]



数据可靠性 & 持久化

- Marathon 持久化卷功能，保证ES节点重启之后能在原机器节点之上restart
- ES 每个索引至少有一个备份 (`index.number_of_replicas>=1`)
- 默认配备hdfs插件，可将数据备份到hdfs上
- 每个宿主 (mesos slave) 上，同一ES集群的节点数不超过索引备份数 (防止主备分片被分配在同一台机器上面)

自动化配置

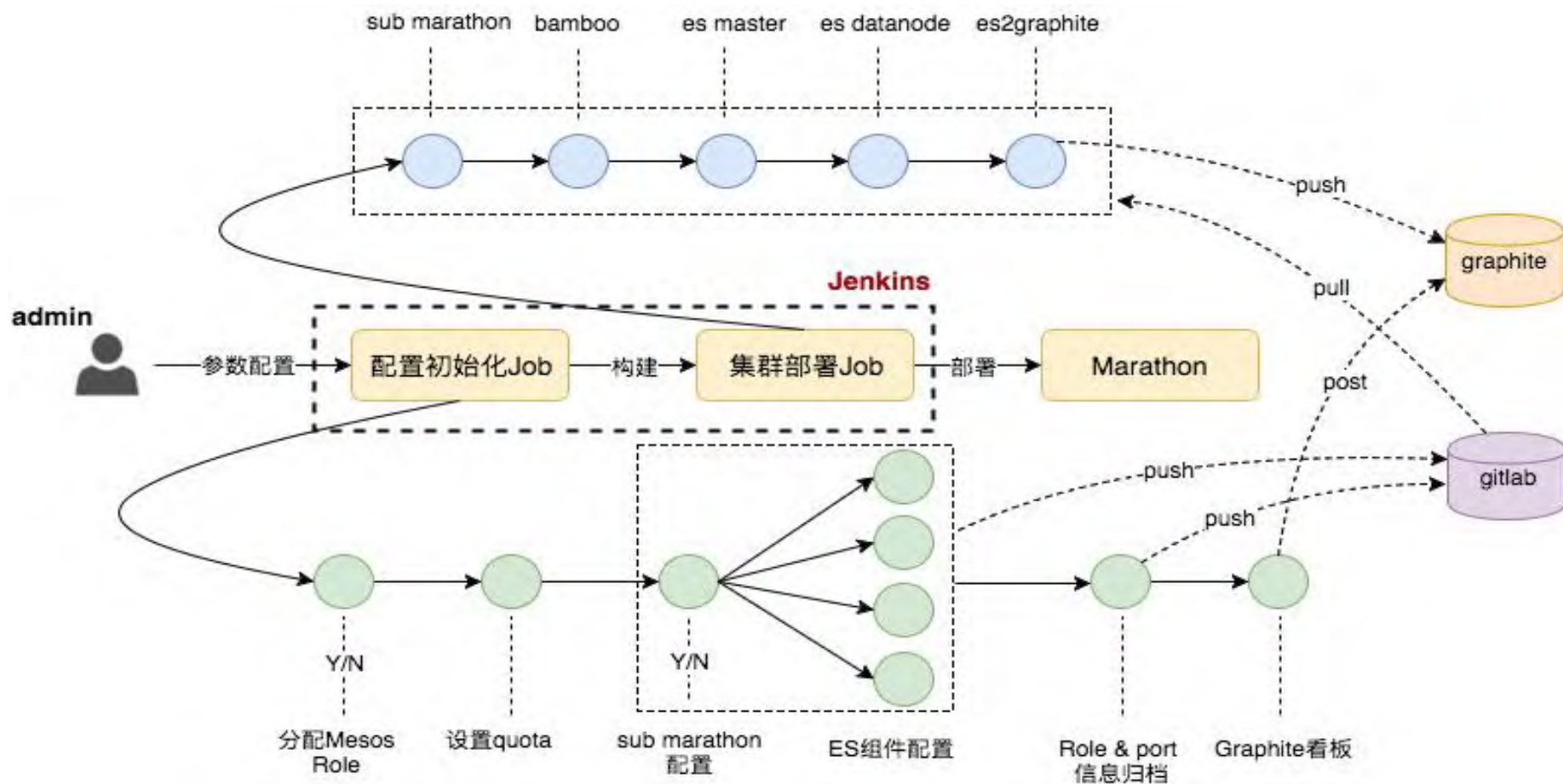


集群信息展示 & 配置

The screenshot displays a web interface for cluster management. On the left, there is a sidebar with navigation options: 'ops_event_log_qes', '集群概况', '集群配置', and '操作日志'. The main content area shows the '操作日志' (Operation Log) for the 'ops_event_log_qes' cluster, which is in a 'Health' state. The log table contains the following entries:

时间	用户	变更url	变更内容
2016-11-14 14:09:13	wen.ma	/install_plugin	install success
2016-11-14 14:09:05	wen.ma	/edit_switch	edit switch status: True
2016-11-03 14:47:49	wen.ma	/restart_node	restart node success
2016-11-03 14:46:09	wen.ma	/restart_node	restart node success
2016-11-03 14:44:11	wen.ma	/restart_node	restart node error: restart_node() takes no arguments (2 given)
2016-11-03 14:43:57	wen.ma	/edit_switch	edit switch status: True
2016-10-31 14:55:04	wen.ma	/edit_switch	edit switch status: False
2016-10-31 14:54:39	wen.ma	/cache_clear	cache clear success
2016-10-31 14:54:34	wen.ma	/edit_switch	edit switch status: True
2016-10-09 12:28:34	wen.ma	/login	login
2016-10-09 12:15:04	wen.ma	/login	login
2016-09-27 10:51:08	songkuan.zheng	/login	login
2016-09-23 10:37:06	wen.ma	/install_plugin	install error: install kibana error: ["message": "An app with id (/ops-event-log-qes/kibana) already exists."]
2016-09-21 11:35:39	tianfan.lien	/login	login
2016-09-18 19:49:48	xulei.xu	/login	login
2016-09-12 19:17:18	congkue.zhang	/login	login
2016-09-06 16:00:43	yun.huang	/login	login
2016-09-06 15:41:05	wen.ma	/edit_switch	edit switch status: True
2016-09-02 14:27:32	yun.huang	/login	login
2016-09-02 12:23:16	wen.ma	/login	login

自动化部署



定制的Jenkins

The screenshot displays the Jenkins web interface for a job named 'Qunar Build'. The interface is divided into several sections:

- Navigation:** A sidebar on the left contains links for 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build with Parameters', 'Delete Project', 'Configure', 'Rebuild Last', 'Job Config History', and 'Qunar Build'.
- Build History:** A section below the navigation sidebar shows a list of recent builds. Each entry includes a build number (e.g., #5, #4, #3, #2, #1), a timestamp (e.g., Aug 15, 2016 2:34 PM), and the build user/environment (e.g., build_user:wen.ma, env_name:test). A search bar is present above the list, and RSS feeds for 'all' and 'failures' are at the bottom.
- Build settings:** The main configuration area on the right is titled 'Build settings'. It includes a '[全选][反选]' button and a list of checkboxes for selecting build modules. The 'ops_event_log_qes_masternode' checkbox is checked. Below the checkboxes, several fields are filled with values:
 - `marathon_url`: `http://marathon_ops-opsdev-qsaa.marathon.corp.qunar.com`
 - `build_Method`: `custom_marathon`
 - `auth`: `Basic: YWRtaW46YWRtaW4=`
 - `git_Root`: `git@gitlab.corp.qunar.com:esaascon/carbon/ops_event_log_qes.git#elasticsearch/es_master.json`
 - `deploy_dependency`: `ops_event_log_qes_bamboo`
- Buttons:** A blue button labeled '点击开始构建发布' (Click to start building and publishing) is located at the bottom of the 'Build settings' section.

监控与报警

- pyadvisor 收集docker 容器的指标
- es2graphite 收集ES集群的指标

指标聚合汇总



报警设置

- ES集群
 - ES集群状态（非green状态）
 - ES节点GC时间
 - 节点数量
- ESAAS平台
 - ZK相关
 - 节点load
 - 节点磁盘容量

日志/事件收集

平台日志分为两个部分：

1. Mesos日志

flume + kafka + logstash + elasticsearch + kibana

1. ES日志

filebeat + kafka + logstash + elasticsearch + kibana

Marathon事件收集：

1. nodejs 收集客户端 + kafka + logstash + elasticsearch + kibana

开发中的问题

- Mesos slave重启之后原有task不会原地恢复
 - Slave id会改变, mesos认为是新的slave
 - boot_id 标识机器是否重启
- Marathon (1.1.1) 持久化卷bug
 - 宿主持久化目录mount错误
 - 持久化目录不能自定义
 - 只使用持久化特性, 自行挂载数据卷作为持久化卷
- 使用持久化卷特性, mesos slave只能root启动
 - 需要mount数据卷
 - ES/marathon 使用非root用户启动

运维中的问题

- 局部资源过热
 - I/O, load, 磁盘
 - 统计分析, 标准差来衡量, 平衡遵从原则: 先手动后自动
- ES的JVM内存
 - 内存管理不是很好, 5.X之前CMS, 之后G1
 - 做好监控&报警, 做好规划, 及时扩容
 - Query/Filter Cache
 - Fielddata Cache
 - Index Buffer
 - segment是JVM old memory消耗的主要者
 - 与数据量成正比, 定期Force merge (索引越大越不明显)
 - Index.max_result_window, size巨大可导致内存被打爆

平台规模

- 平台机器数量： 87台服务器
- datanode数据节点机器数量： 79台服务器
- 当前托管的集群数量： 48个
- 当前可存储数据总量： 200TB左右
- 当前覆盖业务线： 30 个
- 最大的集群数据量： ~ 50 T

Thank you !

