



第九届中国系统架构师大会
SYSTEM ARCHITECT CONFERENCE CHINA 2017

统一资源调度平台建设实践

许令波/2017.10

SACC
2017

北京·新云南皇冠假日酒店



内容简介

- 为什么需要资源调度
 - 什么是调度
 - 提升稳定性和效率
 - 建设数据中心
- 如何抽象资源
 - 资源的收集和管理。
 - 虚拟化
 - 数据持久化
- 如何调度资源
 - 技术选型
 - 兼容考虑
- 遇到的难题

SACC2017

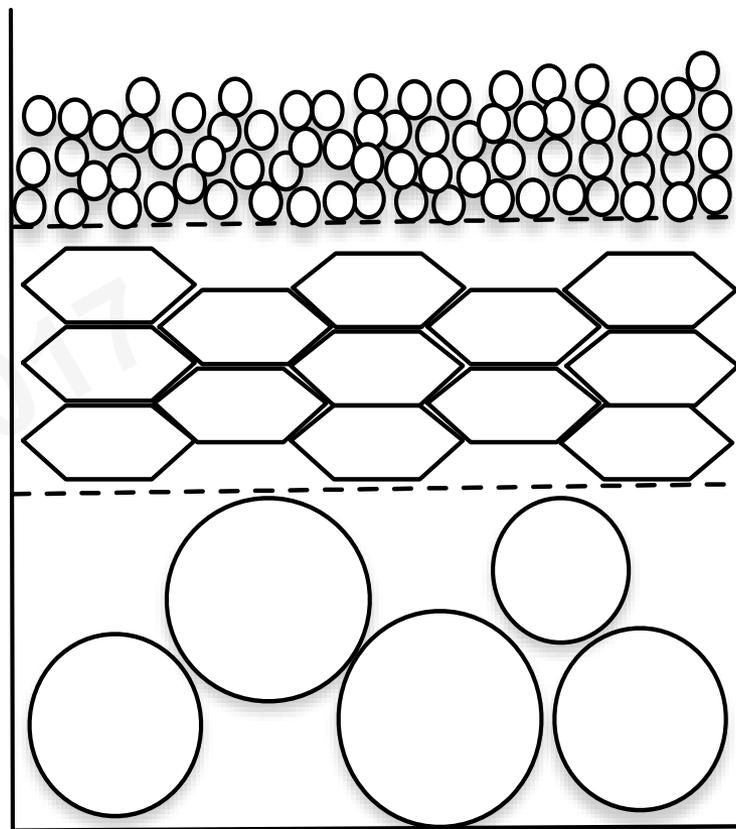
关于我

- 联系我: xulingbo0201@163.com
- 目前负责滴滴的资源调度平台建设, 主要关注资源调度和容器化相关技术
- 之前有在阿里7年的性能优化经验
- 著有《深入分析Java Web技术内幕》

什么是调度

所谓资源调度一般分为两个阶段：

- 虚拟化（即资源的抽象）：虚拟机或容器技术来隔离资源
- 编排：更细粒度在时间和空间上优化资源的使用



一些数据

- 从物理机迁入oceanbank,一台GIFT-PROXY可省：22核，120G内存。一台GIFT-FS可省9核CPU，40G内存
- 节约成本统计：50多台GIFT-PROXY和60多台GIFT-FS共节约500多核，2T内存。折算成物理机共省下约：节省了约50% M机型物理机。

提升稳定性和效率

■ 提升运维效率

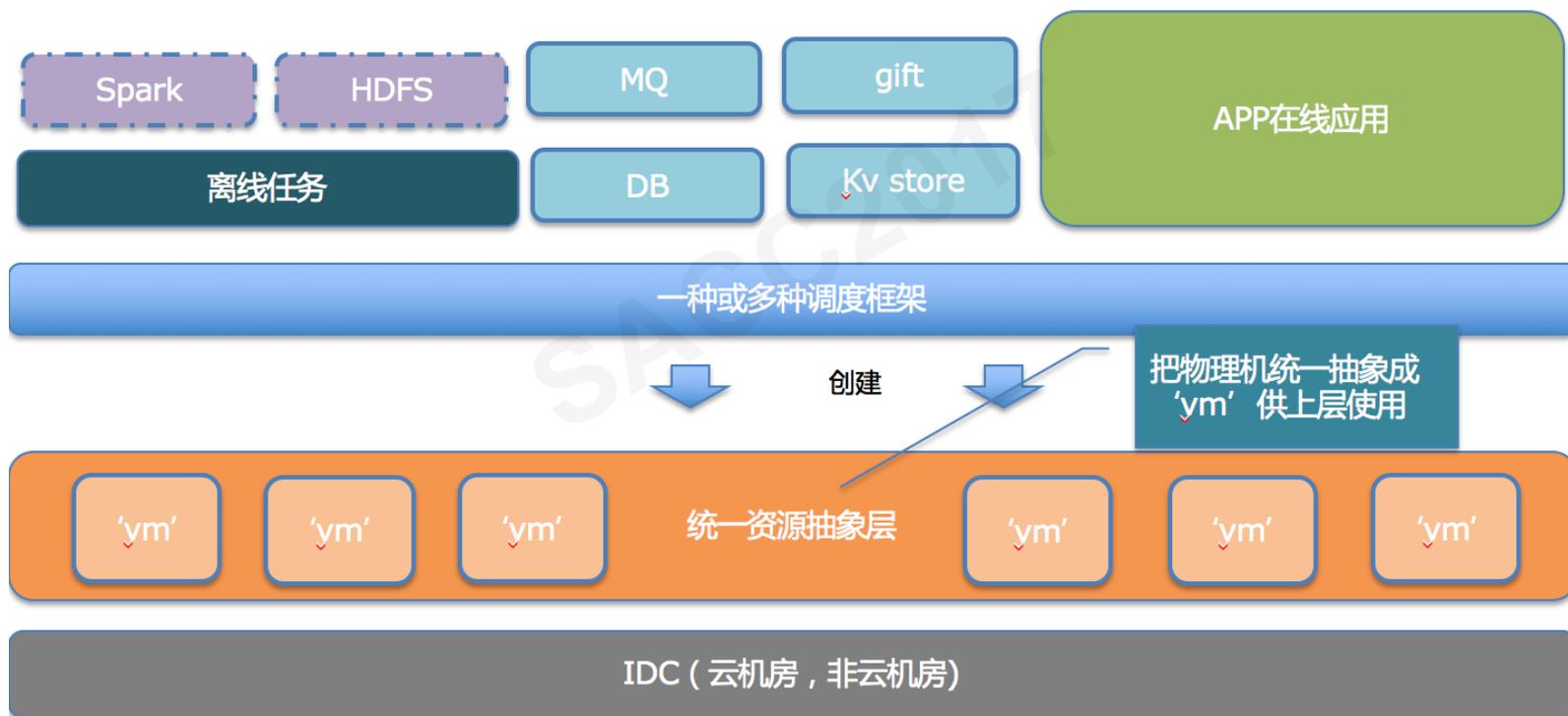
- 标准化运维，只要一个镜像就可以部署起来
- 自动化，应用依赖和部署变自动化减少了人为的干预

■ 提升稳定性：

- 应用和运行的机器解耦了
- 相应的就可以做弹性伸缩和调度

建设数据中心

资源的统一抽象对上层应用屏蔽时间（不会down机）和空间（机器、机房、区域等）上的差异



如何抽象资源

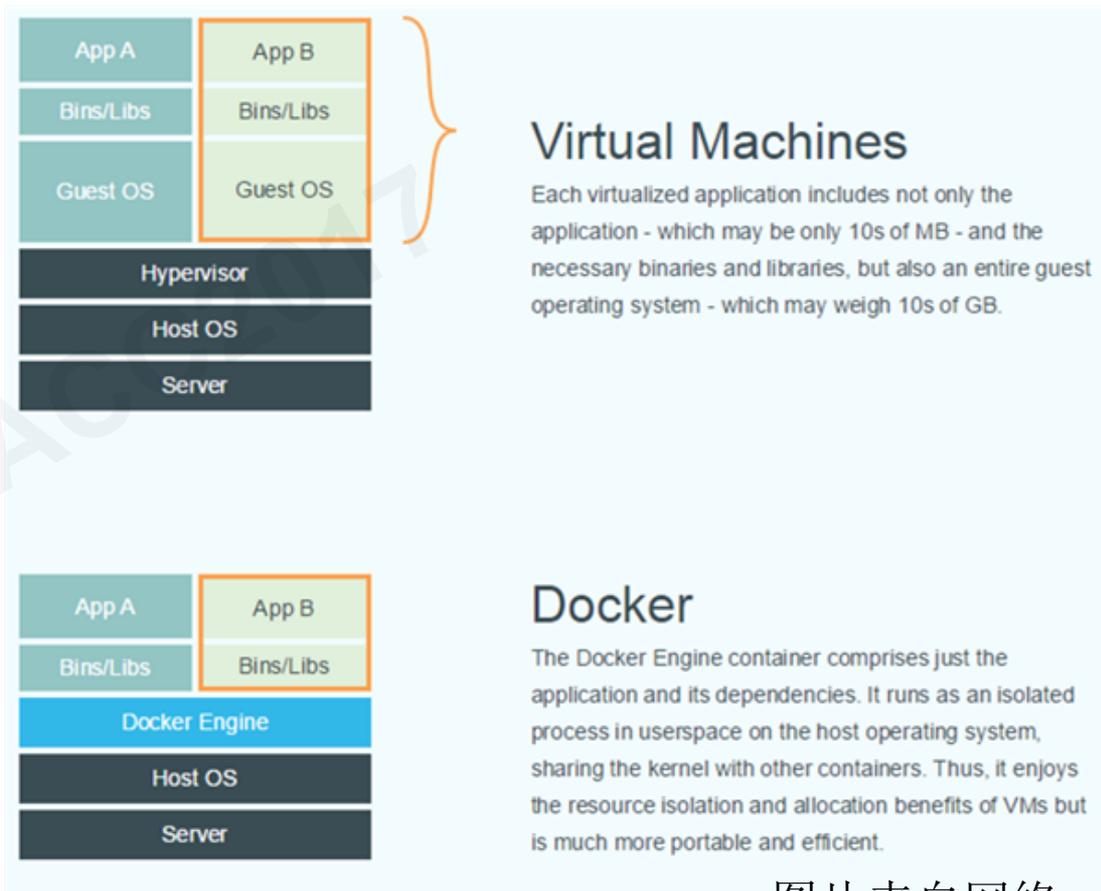
- 对具体物理资源的收集和管理。
- 虚拟化。将抽象的硬件资源属性进行重新封装，封装成上层可以使用的一个实体，可以是容器也可以是虚拟机或者一个资源集合。
- 数据持久化。业务少不了会有数据存储在本机，这样就会把机器变成有状态，不利于对资源的全局调度，所以有三种场景需要解决：一是不需要永久本地存储但是会实时写到本地如应用的日志；二是需要永久存储如DB。三是分布式存储场景，做到存储与计算分离

物理资源的收集和管理

- 资源的信息管理。有多少，用了多少，还有多少。
- 大量物理机器的集群管理。除了通常几十万台的管理功能外，还有一分部的任务管理，如负责接收master的创建容器的task等
- 资源的合理分配策略和算法。上层的资源请求最终会在每台的物理机上进行分配以及如何分配，这里面有很多优化空间，如何保证每台物理机资源的合理利用需要合理的分配策略和算法支撑。

虚拟化

vm、docker和lxc的选择问题:

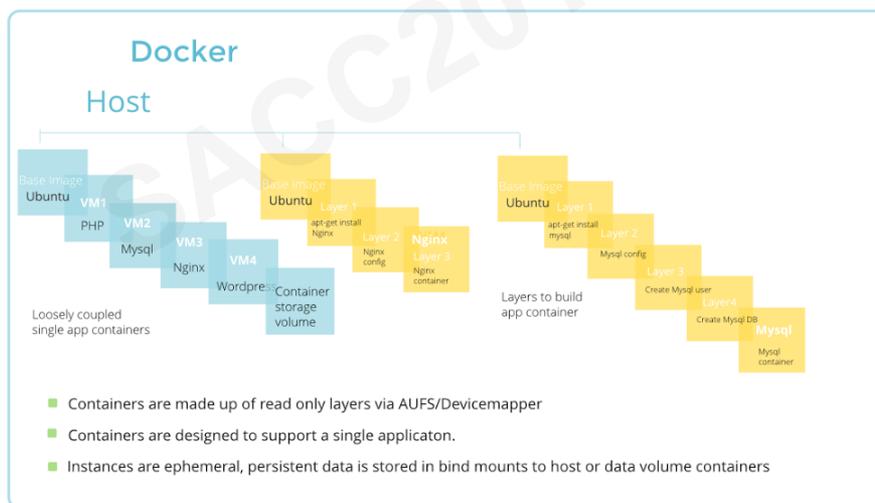
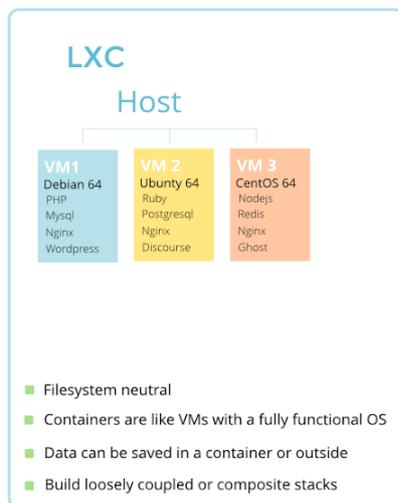


图片来自网络

Docker vs LXC

- copy on write技术可以节省磁盘
- 非volume挂载磁盘会有性能损耗

Key differences between LXC and Docker

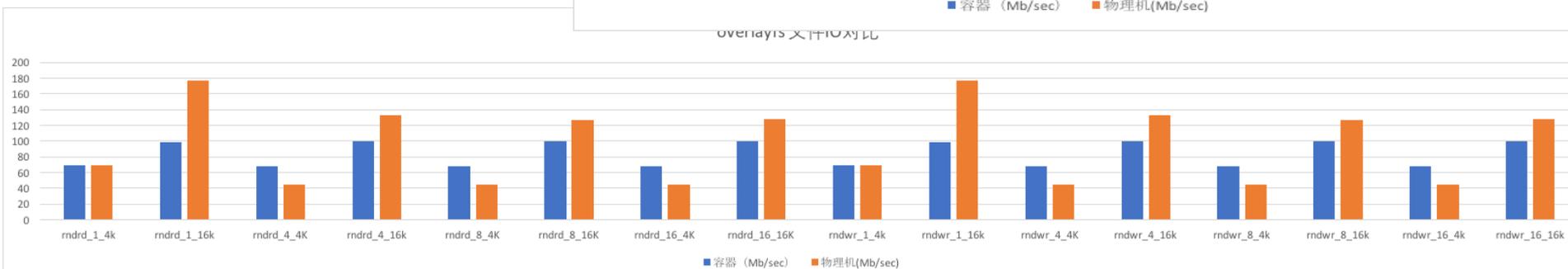
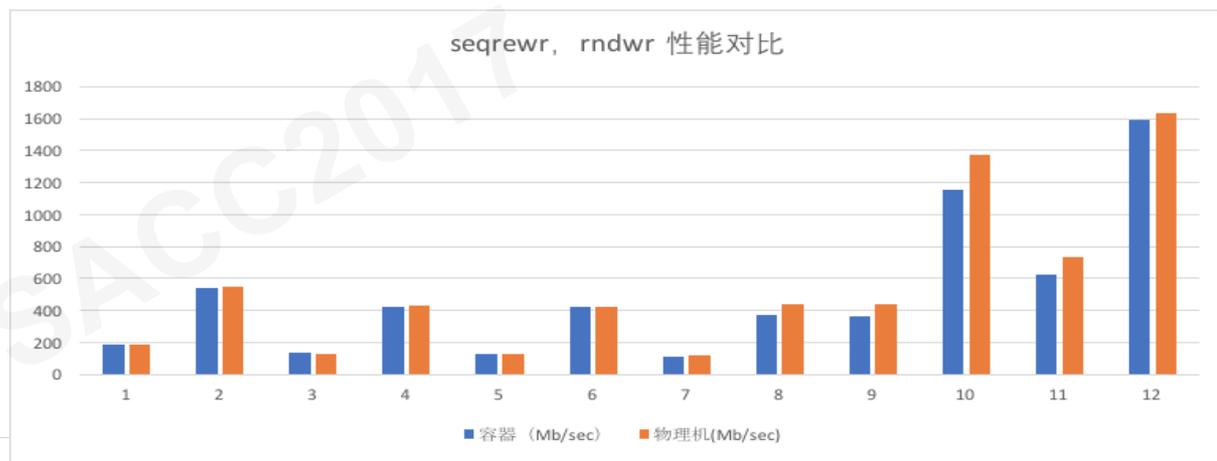


图片来自网络

容器存储选型

■ Devicemapper性能对比

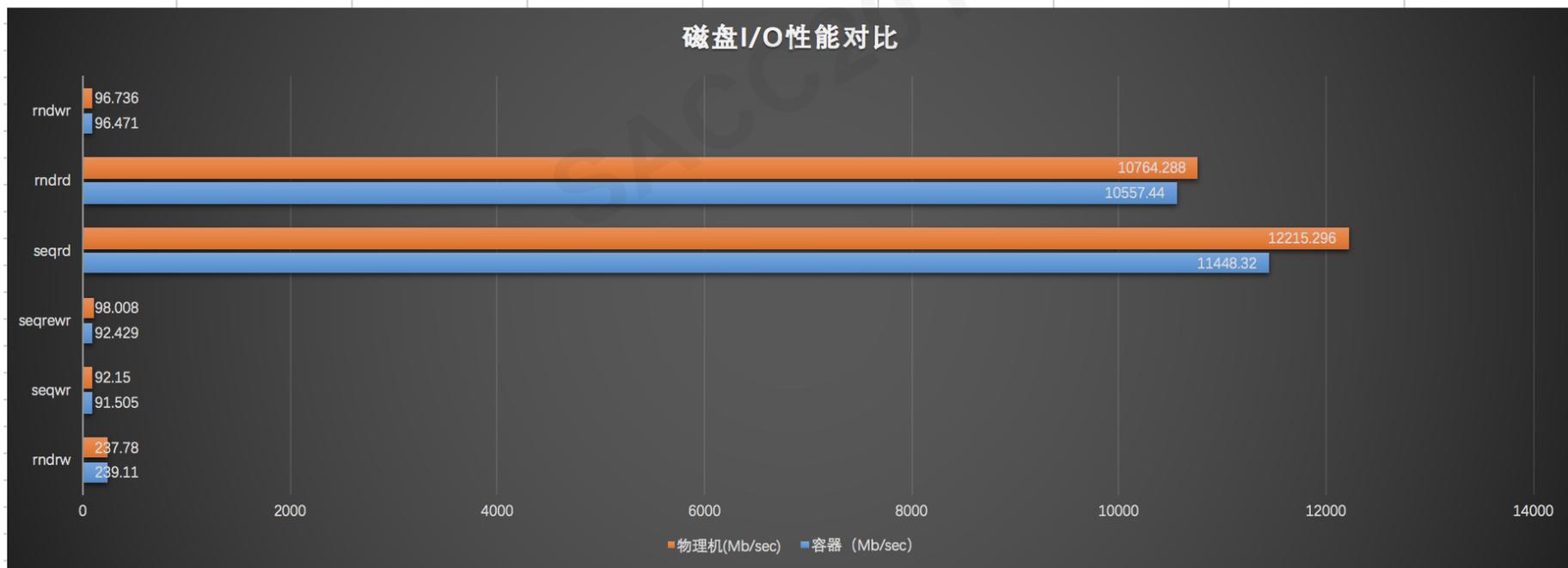
■ OverlayFS性能对比



LXC的磁盘性能

■ LXC与宿主机磁盘I/O性能测试对比

序号	磁盘读写方式	容器 (Mb/sec)	物理机(Mb/sec)	容器(Requests/sec)	物理机(Requests/sec)	并发	块大小 (字节)	文件大小 (G)
1	rndrw	239.11	237.78	15292.89	15217.78	16	16384	3
2	seqwr	91.505	92.15	5856.34	5897.61	16	16384	3
3	seqrewr	92.429	98.008	5915.48	6272.49	16	16384	3
4	seqrd	11448.32	12215.296	732709.18	781810.57	16	16384	3
5	rndrd	10557.44	10764.288	675648.2	688908.61	16	16384	3
6	rndwr	96.471	96.736	6174.13	6191.1	16	16384	3



如何调度资源

- 调度框架选型
- 调度策略

SACC2017

调度框架选型

■ K8s mesos swarn等比较

	Framework	Architecture	Resource granularity	Multi-scheduler	Pluggable logic	Priority preemption	Re-scheduling	Oversubscription	Resource estimation	Avoid interference
O P E N	Kubernetes	monolithic	multi-dimensional	N ^[v1.2, DD, Issue]	Y ^[DD]	N ^[Issue]	N ^[Issue]	Y ^[DD]	N	N
	Swarm	monolithic	multi-dimensional	N	N	N ^[Issue]	N	N	N	N
	YARN	monolithic/ two-level	RAM/CPU slots	Y	N ^[app-lvl. only]	N ^[IIRA]	N	N ^[IIRA]	N	N
	Mesos	two-level	multi-dimensional	Y	Y ^[framework-lvl.]	N ^[IIRA]	N	Y ^[v0.23, Doc]	N	N
	Nomad	shared-state	multi-dimensional	Y	Y	N ^[Issue]	N ^[Issue]	N ^[Issue]	N	N
	Sparrow	fully-distributed	fixed slots	Y	N	N	N	N	N	N
C L O S E D	Borg	monolithic ^[2]	multi-dimensional	N ^[2]	N ^[2]	Y	Y	Y	Y	N
	Omega	shared-state	multi-dimensional	Y	Y	Y	Y	Y	Y	N
	Apollo	shared-state	multi-dimensional	Y	Y	Y	Y	N	N	N

图片来自网络

我们最关心的几个因素

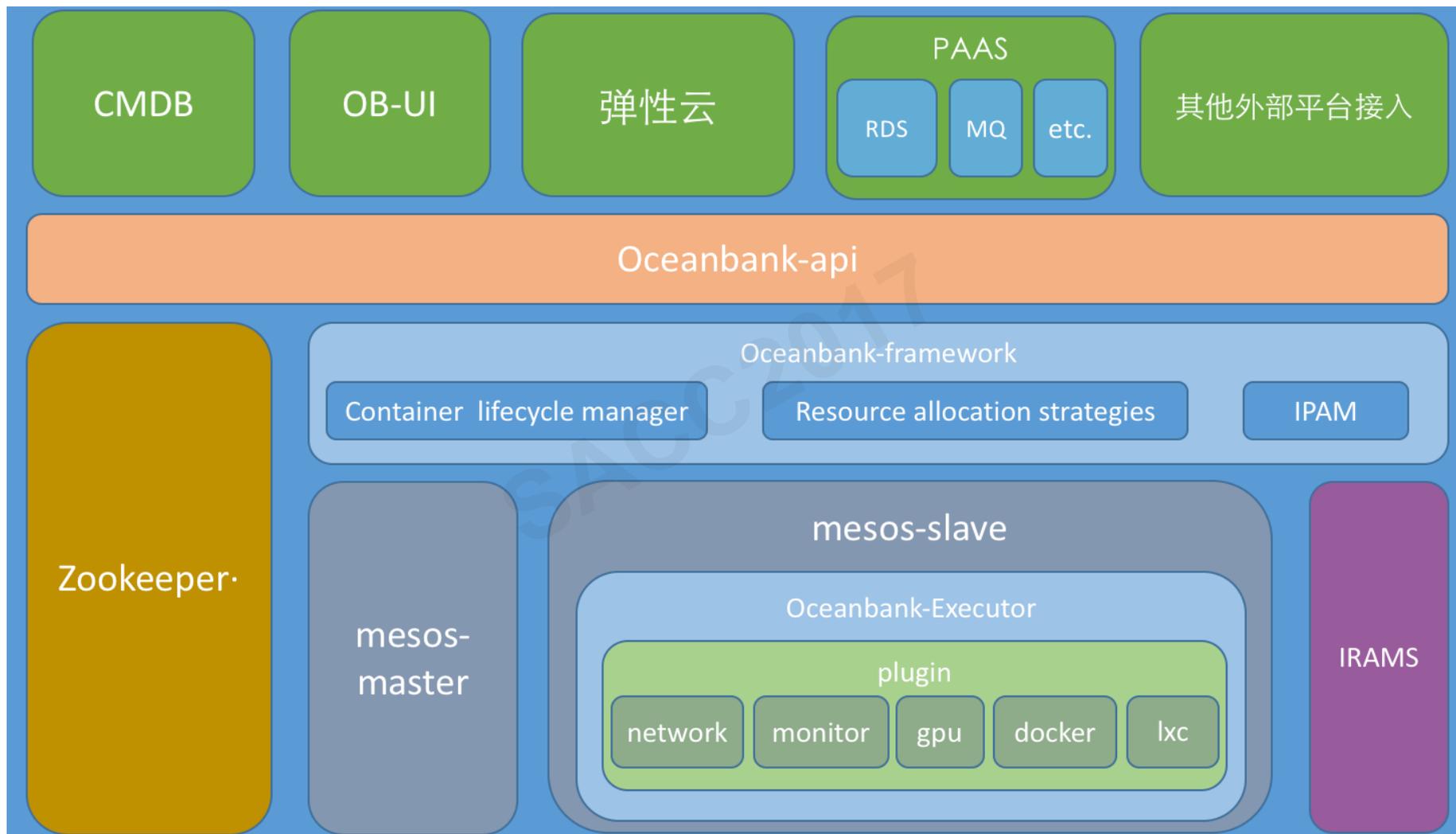
■ 支持的规模、稳定性

- Mesos 支撑 >30000 nodes; 250k containers
- Twiter/Apple/Uber/ebay/airbnb 等

■ 兼容性、接入成本

- 支持大数据集群的接入
- 支持各种中间件的接入

统一调度的架构



统一调度的架构

总览

总览

申请容器

容器列表

说明文档

资源使用情况

物理机总数: 剩余 资源

当前已接入的容器总数

容器套餐 华新园机房 机房

总览

容器列表

虚拟机管理

物理机管理

产品线管理

套餐管理

操作任务管理

容器编号 容器IP 宿主主机名 容器状态

新建

<input type="checkbox"/>	主机名	宿主(物理机)	所属机房	所属产品线	类型	状态	最后操作时间	操作
<input type="checkbox"/>	f53be669d40b	.gz01	华新园机房		docker	运行	2017-10-13 18:17:35	改名
<input type="checkbox"/>	9ba2ec34252d	703.gz01	华新园机房		docker	运行	2017-10-13 18:14:13	改名
<input type="checkbox"/>	765ef908699b	-702.gz01	华新园机房		docker	运行	2017-10-13 15:44:44	改名
<input type="checkbox"/>	d0a2ca64951b	ob-slave-702.gz01	华新园机房		docker	运行	2017-10-13 15:44:44	改名

中间件接入

- 提供容器asVM，将容器做成VM一样的体验
- 利用LXCFS隔离CPU、mem和Load信息

SACC2017

实现LXCFS的load隔离

- 计算公式： $\text{load}(t) = \text{load}(t-1) \exp(-5/60R) + n(t) (1 - \exp(5/60R))$
- 如果想在容器中获取正确Loadavg信息的那么就要具备以下几点：
 - 获取运行在容器中的所有进程（包括：线程）。
 - 获取运行在容器中的进程总数。
 - 获取运行在容器中的所有进程运行状态。
 - Loadavg计算公式。
- 难点是如何控制性能消耗
 - 控制对进程信息获取的系统调用

大数据接入

- 直接提供容器as vm给大数据使用，直接将yarn部署在容器里
- 在离线混部需要考虑磁盘的存储容量问题
- 资源动态调度支持

SACC2017

遇到的坑

1. Java线程夯住的问题排查过程
2. lxc容器cgroup被修改问题分析
3. 【源码分析】LXC存在配置文件解析的问题
4. Ob系统内存分配失败oom问题分析
5. LXCFS获取proc文件长度失败排查

示例2- Ob系统内存分配失败问题分析

```
[7180785.639634] odin-metrics invoked oom-killer: gfp_mask=0xd0, order=0, oom_score_adj=0
[7180785.640772] odin-metrics cpuset=e9605ef9cf58d248872a2f9c2992d26e0f8b299b923bb4705f5f4c7cfef9b02c mems_allowed=0-1
[7180785.641817] CPU: 21 PID: 19149 Comm: odin-metrics Tainted: G      W OE K----- 3.10.0-514.16.1.el7.x86_64
[7180785.642859] Hardware name: INSUR SA5212M4/Shuyu, BIOS 4.0.7 03/29/2016
[7180785.642864] ffff880bb2e89f60 000000005cb60e37 ffff880100aefcc0 ffffffff81686ac3
[7180785.642865] ffff880100aefd50 ffffffff81681a6e ffff8804d5ff9d80 0000000000000001
[7180785.642866] 0000000000000000 0000000000000000 000000000000000046 ffffffff811847c6
[7180785.642867] Call Trace:
[7180785.642874] [<ffffffff81686ac3>] dump_stack+0x19/0x1b
[7180785.642880] [<ffffffff81681a6e>] dump_header+0x8e/0x225
[7180785.642886] [<ffffffff811847c6>] ? find_lock_task_mm+0x56/0xc0
[7180785.642889] [<ffffffff81184c7e>] oom_kill_process+0x24e/0x3c0
[7180785.642891] [<ffffffff8118471d>] ? oom_unkillable_task+0xcd/0x120
[7180785.642896] [<ffffffff81093c0e>] ? has_capability_noaudit+0x1e/0x30
[7180785.642899] [<ffffffff811f3651>] mem_cgroup_oom_synchronize+0x551/0x580
[7180785.642901] [<ffffffff811f2a0b>] ? mem_cgroup_charge_common+0xc0/0xc0
[7180785.642904] [<ffffffff81185504>] ?
[7180785.642905] [<ffffffff81167f8da>] system.journal
[7180785.642909] [<ffffffff81692885>] system0000555a54c4a84db-bbfee6cc48d586fb.journal-
[7180785.642944] [<ffffffffffa0626c99>] system0000555a57e6af5ae-af32585f9bdaf2f3.journal-
[7180785.642946] [<ffffffffff81692975>] system0000555a5a5b8e743-576cc325e4e0a08c.journal-
[7180785.642948] [<ffffffffff8168eb88>] system0000555ab11589ab2-78257f546732ea5.journal-
[7180785.642951] Task in /docker/e96055b5f54c7cfef9b02c
[7180785.642952] memory: usage 4194304
[7180785.642952] memory+swap: usage 4194304
[7180785.642953] kmem: usage 0kB, 4194304
[7180785.642969] Memory cgroup stats for /docker/e96055b5f54c7cfef9b02c:
B inactive_anon:1487408KB active_anon:1487408KB
[7180785.642969] [ pid ] uid tgid tpid
[7180785.643138] [ 5121 ] 0 5121
[7180785.643141] [19654] 81 19654
[7180785.643143] [19761] 0 19761
[7180785.643145] [20065] 0 20065
[7180785.643146] [22312] 0 22312
```

filename	size	total_pages	min_cached_page	cached_pages	cached_size	cached_perc
system.journal	240	1	0	1	4,096	100.00
system0000555a54c4a84db-bbfee6cc48d586fb.journal-	67,108,864	16,384	0	14,579	59,575,584	88.98
system0000555a57e6af5ae-af32585f9bdaf2f3.journal-	50,331,648	12,288	0	9,131	37,400,576	74.31
system0000555a5a5b8e743-576cc325e4e0a08c.journal-	33,554,432	8,192	0	6,567	26,898,432	80.16
system0000555ab11589ab2-78257f546732ea5.journal-	25,165,824	6,144	0	4,391	17,985,536	71.47
system0000555ab11589ab2-78257f546732ea5.journal-	25,165,824	6,144	0	4,894	20,045,824	79.65
system0000555b5f54c7cfef9b02c	50,331,648	12,288	0	9,859	40,382,464	80.23
system0000555b5f54c7cfef9b02c	16,777,216	4,096	0	2,851	11,677,696	69.60
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,442	67,346,432	89.20
system0000555b5f54c7cfef9b02c	83,886,080	20,480	0	17,035	69,775,360	83.18
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,520	67,665,920	89.63
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,528	67,698,688	89.67
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,479	67,497,984	89.40
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,241	66,323,136	88.11
system0000555b5f54c7cfef9b02c	75,497,472	18,432	0	16,020	65,617,920	86.91
system0000555b5f54c7cfef9b02c	92,274,688	22,528	0	20,977	85,921,792	93.12
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,221	95,113,216	75.59
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,207	95,055,872	75.54
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,271	95,318,016	75.75
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,209	95,064,064	75.55
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,190	95,105,024	75.58
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,218	95,100,928	75.58
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,260	95,272,960	75.72
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,266	95,297,536	75.74
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,237	95,178,752	75.64
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,274	95,330,304	75.76
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,190	95,023,104	75.52
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,258	95,264,768	75.71
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,173	94,916,608	75.43
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,202	95,035,392	75.53
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,203	95,039,488	75.53
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,190	94,986,240	75.49
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,209	95,064,064	75.55
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,151	94,826,496	75.46
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,189	94,982,144	75.49
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,220	95,109,120	75.59
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,161	94,867,456	75.39
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,177	95,096,832	75.58
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,177	94,932,992	75.45
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,198	95,019,008	75.51
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,194	95,003,624	75.50
system0000555b5f54c7cfef9b02c	125,829,120	30,720	0	23,216	95,092,736	75.57

total_cached_size: 3,244,253,184

THANKS

The background features a dark, almost black, space filled with numerous bright blue particles. These particles are arranged in several distinct, curved paths that sweep across the frame from the bottom left towards the top right. A bright, white-to-yellow light source is positioned near the center, casting a soft glow and illuminating the particles closest to it. The overall effect is one of dynamic energy and digital connectivity.