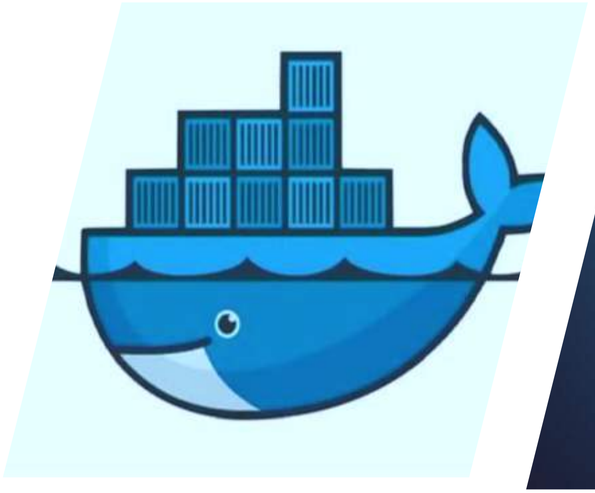
QCon 全球软件开发大会
INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

BEIJING 2018

容器云在头条的落地和实践

演讲者 / 郑建磊

主办方 Geekbang. 极客邦科技 InfoQ

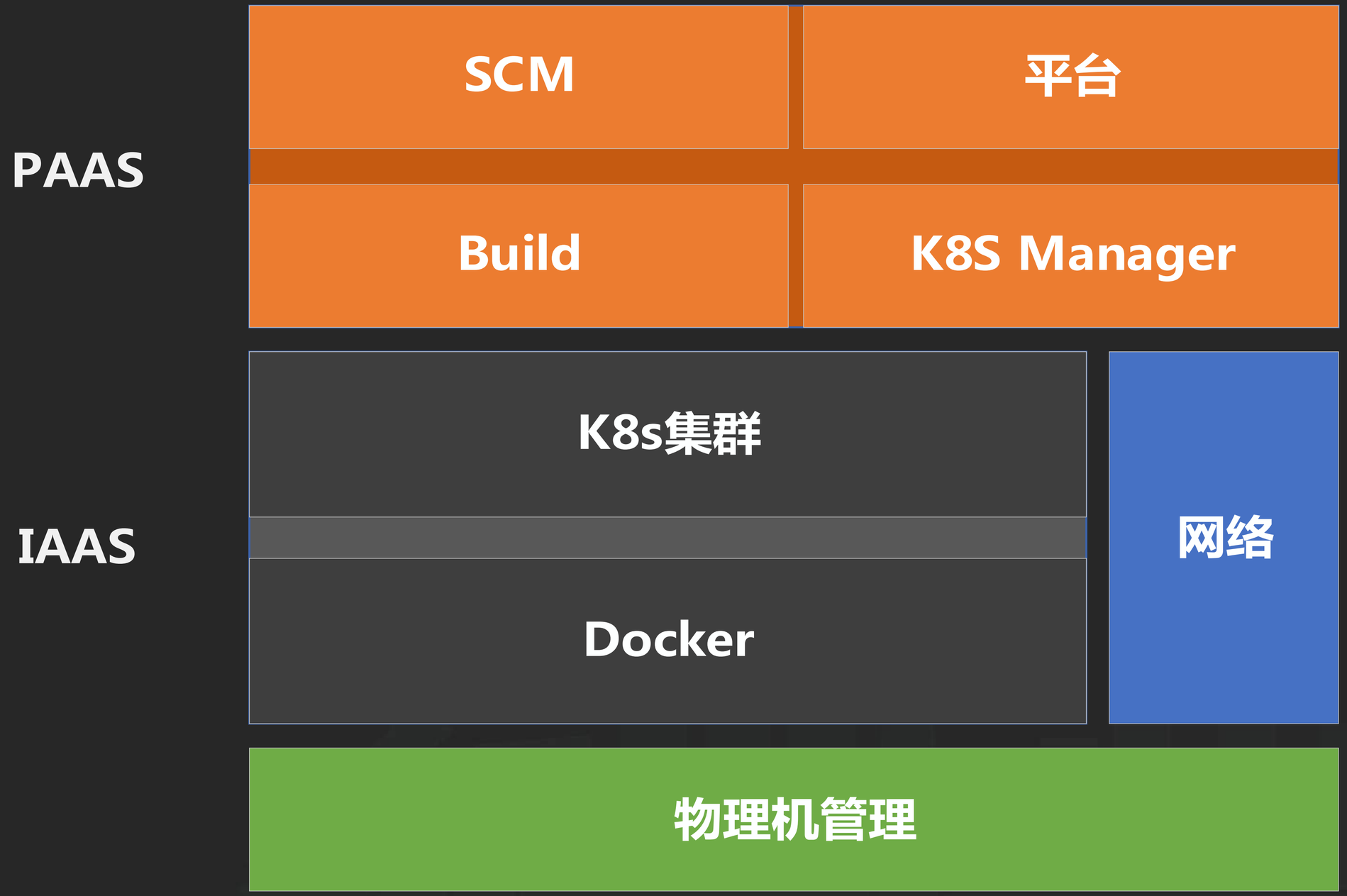


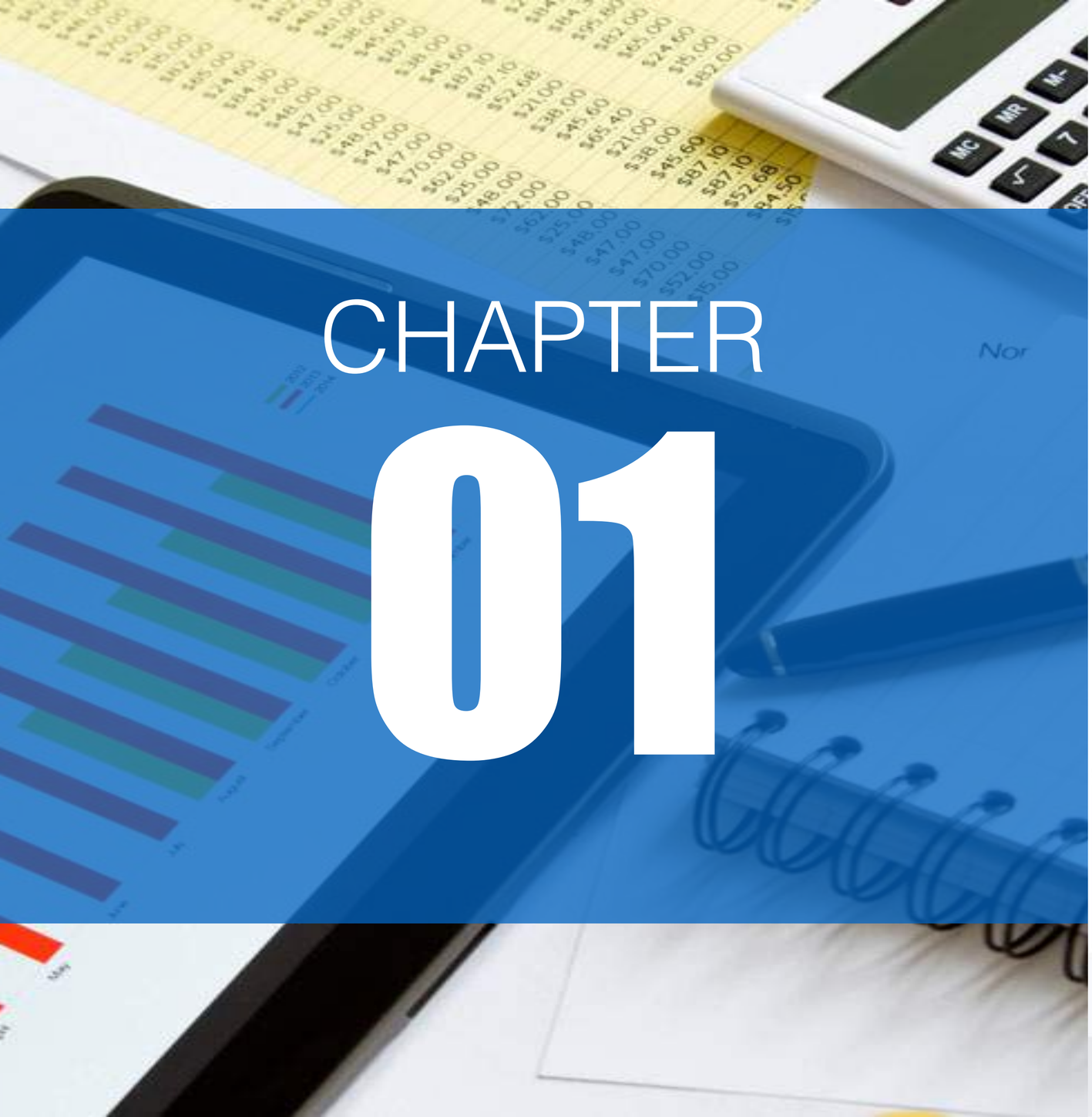
TCE

(Toutiao Compute Engine)

目录

CONTENTS

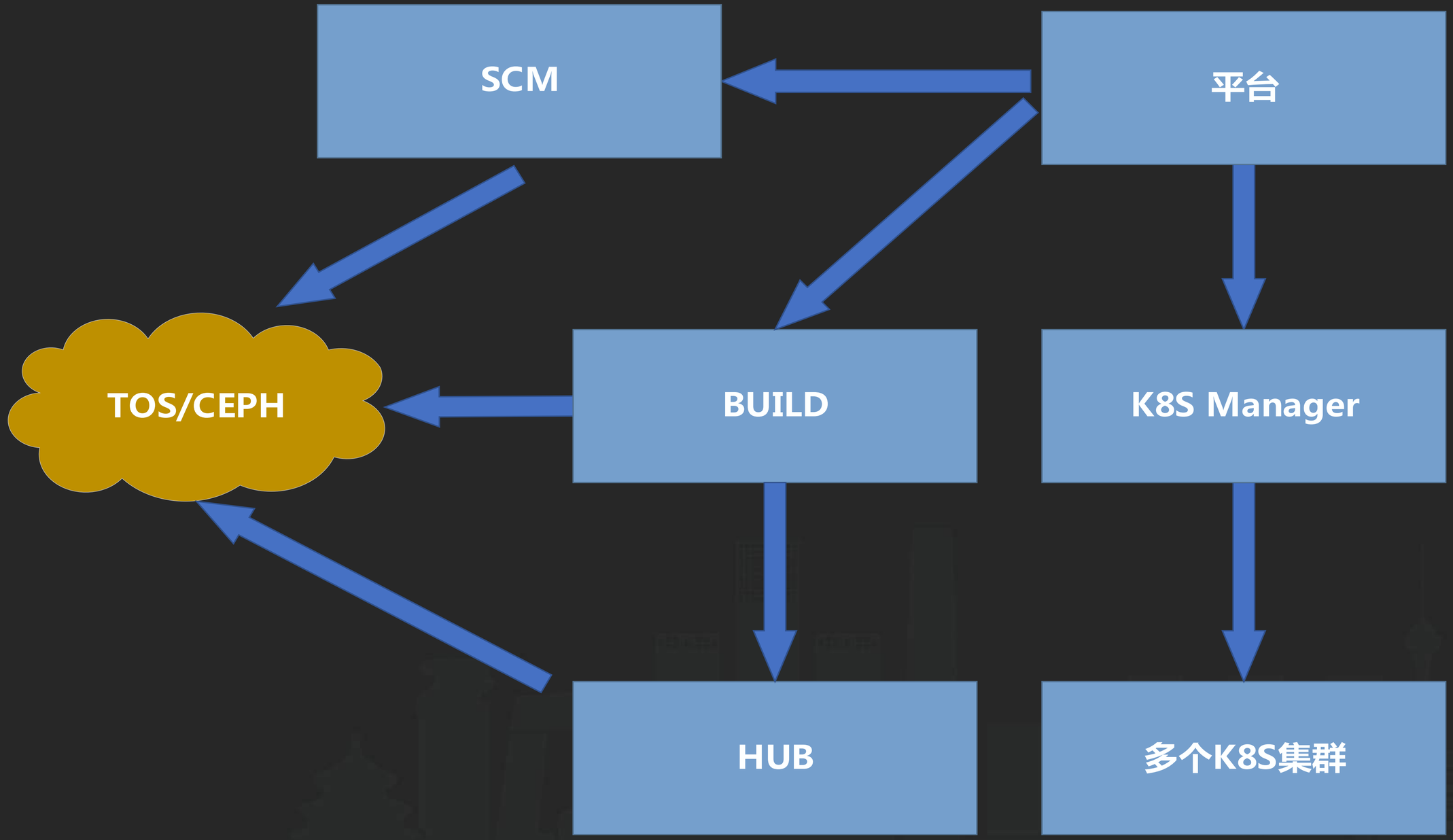

PAAS
SCM
平台
Build
K8S Manager
IAAS
K8s集群
网络
Docker
物理机管理



CHAPTER

01

PAAS


SCM
平台
TOS/CEPH
BUILD
K8S Manager
HUB
多个K8S集群

SCM


toutiao/demo/http_demo
类型：go1.10
创建者：
SSH
/http_demo.git
+ 发布
配置
版本列表
仓库依赖
使用服务
快速链接
云引擎
上线系统
定时任务
短任务
脚本服务
深度学习
[1.0.0.59] add http go demo
布于 2018-04-08 11:10:04
阿里云
上线版本
编译成功
更多
禁用
[1.0.0.58] add http go demo
布于 2018-04-08 11:02:26
阿里云
上线版本
编译成功
更多
禁用
[1.0.0.57] add http go demo
布于 2018-04-08 10:57:35
阿里云
上线版本
编译成功
更多
禁用
[1.0.0.56] add http go demo
布于 2018-04-08 10:53:30
阿里云
上线版本
编译成功
更多
禁用

服务管理

集群信息 | 服务详情 | 服务依赖 | 工单历史 | 变更历史 | 压测历史

服务名称: toutiao.demo

PSM: toutiao.demo.http_demo

所有人: zhengjianlei

所在组: 头条研发-基础架构

基础镜像: toutiao.debian:latest

服务端口: 8889(主端口), 9000(other)

语言类型: go

服务安装脚本: toutiao/app/kdemo

框架类型: none

切片参数:

报警列表:

新建集群时自动设置的metric报警:

服务Tags: -

服务管理

集群信息 | 服务详情 | 服务依赖 | 工单历史 | 变更历史 | 压测历史

依赖仓库	部署路径	当前版本	最新版本	更新
ss_conf	ss_conf	1.0.0.406		
toutiao/demo/http_demo	toutiao/app/kdemo	1.0.0.44		
toutiao/conf	toutiao/conf/	1.0.0.1679		
pyutil	pyutil	1.0.0.91		
toutiao/load	toutiao/load	1.0.1.68		
toutiao/runtime	toutiao/runtime	1.0.0.43		

依赖选装库	安装方式
python-dev	apt-get

服务上线

部署详情

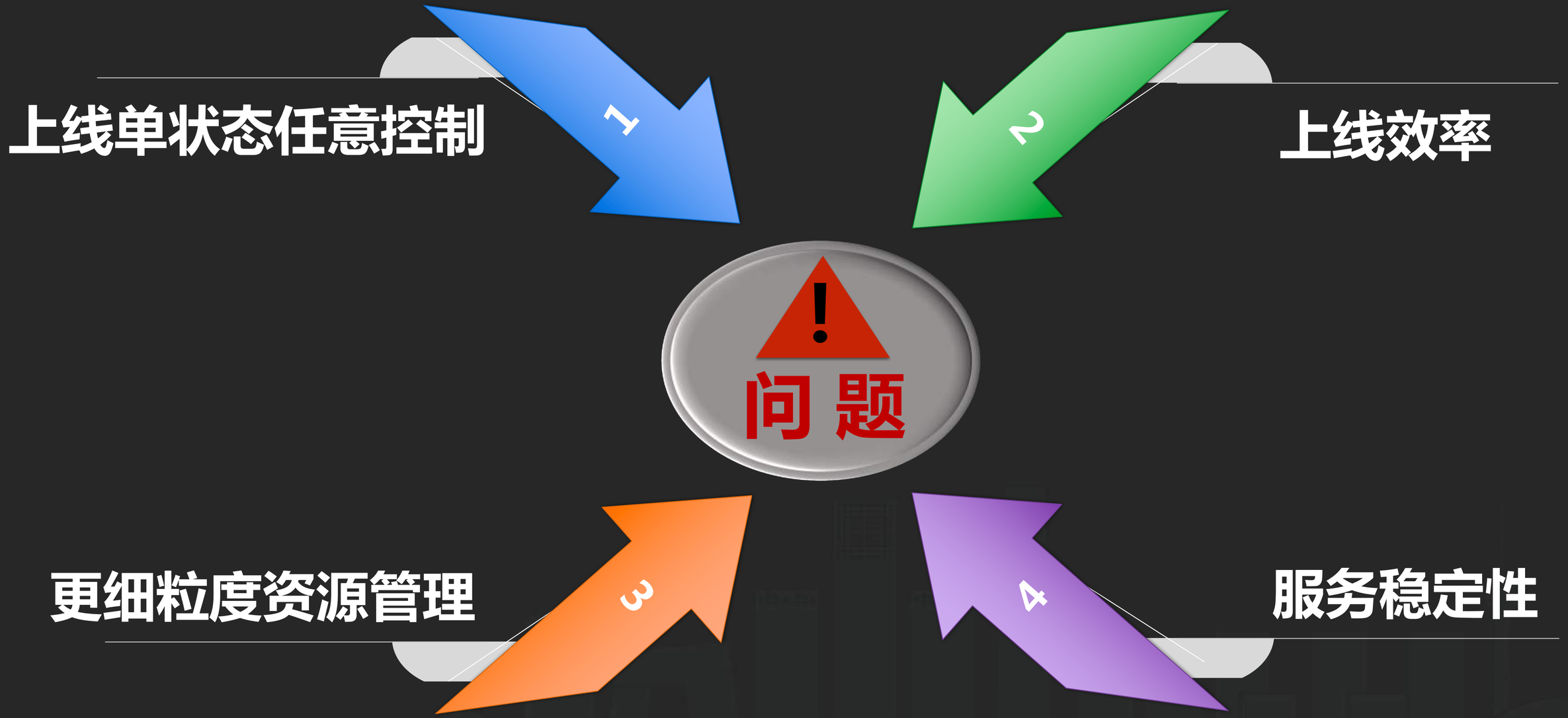
构建 —— 2 部署

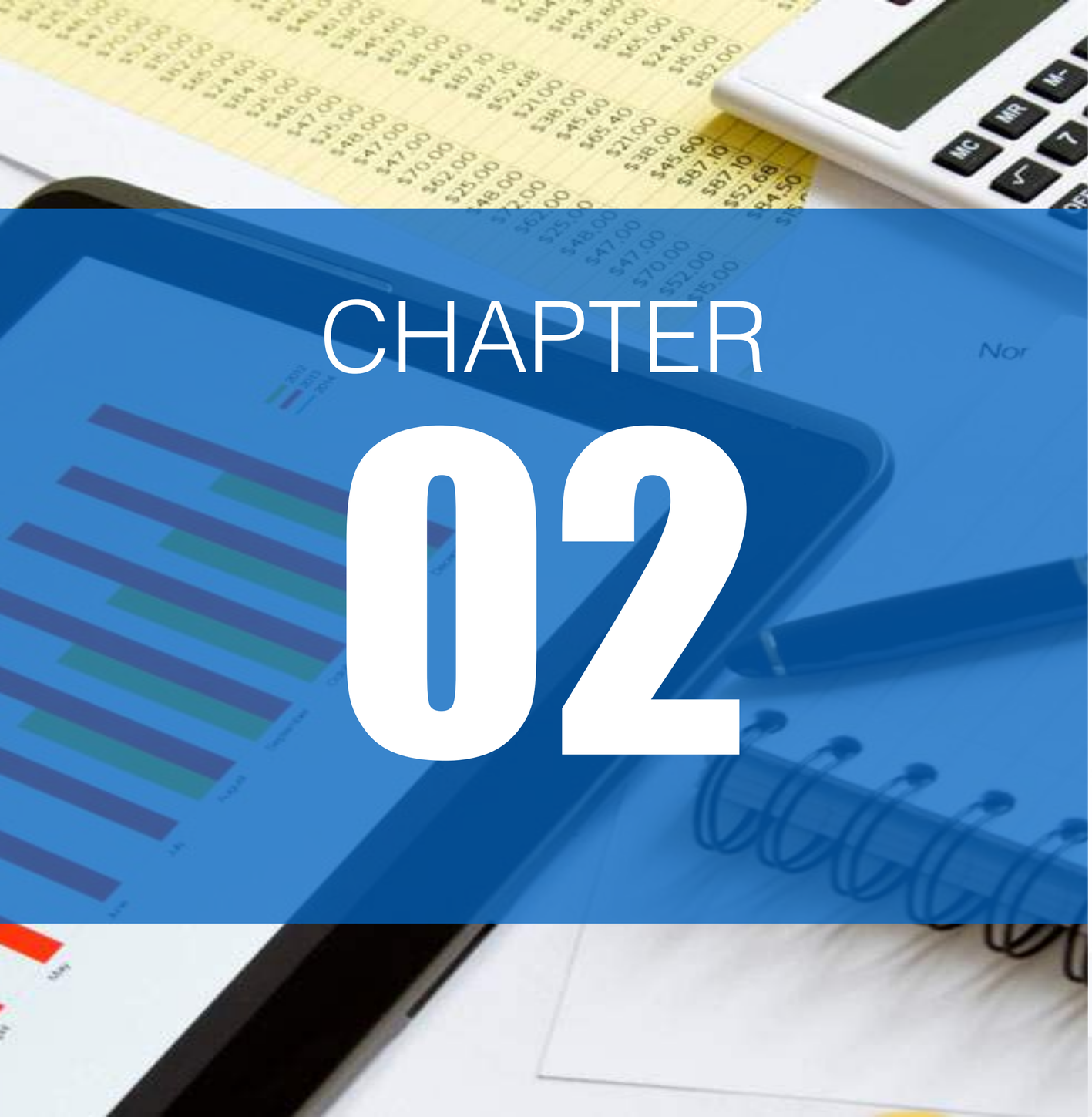
sg-ies-default

步骤名称	状态	开始时间	结束时间	日志	操作
升级 default-小流量 ✓	完成	2018-03-11 11:04:17	2018-03-11 11:05:01	查看日志	
测试 default-小流量 ✓	完成	2018-03-11 11:05:01	2018-03-11 11:05:02	-	
升级 default-单机房 ✓	完成	2018-03-11 11:14:03	2018-03-11 11:14:52	查看日志	

Photo-default

步骤名称	状态	开始时间	结束时间	日志	操作
升级 default-小流量 ✓	完成	2018-03-11 11:04:33	2018-03-11 11:05:02	查看日志	
测试 default-小流量 ✓	完成	2018-03-11 11:05:03	2018-03-11 11:05:04	-	
升级 default-单机房 ✓	完成	2018-03-11 11:46:04	2018-03-11 11:47:05	查看日志	
升级 default-全流量 ✓	完成	2018-03-11 17:01:43	2018-03-11 17:03:07	查看日志	


上线单状态任意控制
1
2
上线效率
问 题
更细粒度资源管理
3
4
服务稳定性



CHAPTER

02

IAAS

K8S层

- 上线单状态任意控制
  - RC -> Deployment
- 上线效率
  - 滚动升级 -> 原地升级
  - cpu超售
  - 抢占式调度
  - 镜像P2P分发 & 预拉取
- 更细粒度资源管理
  - 端口
  - cpuset & numa

Docker层

上线效率

commands such as 'docker run' and 'docker ps' appear to hang indefinitely due to huge request backlog (congestion) in containerd

dockerd leaks ExecIds on failed exec -i

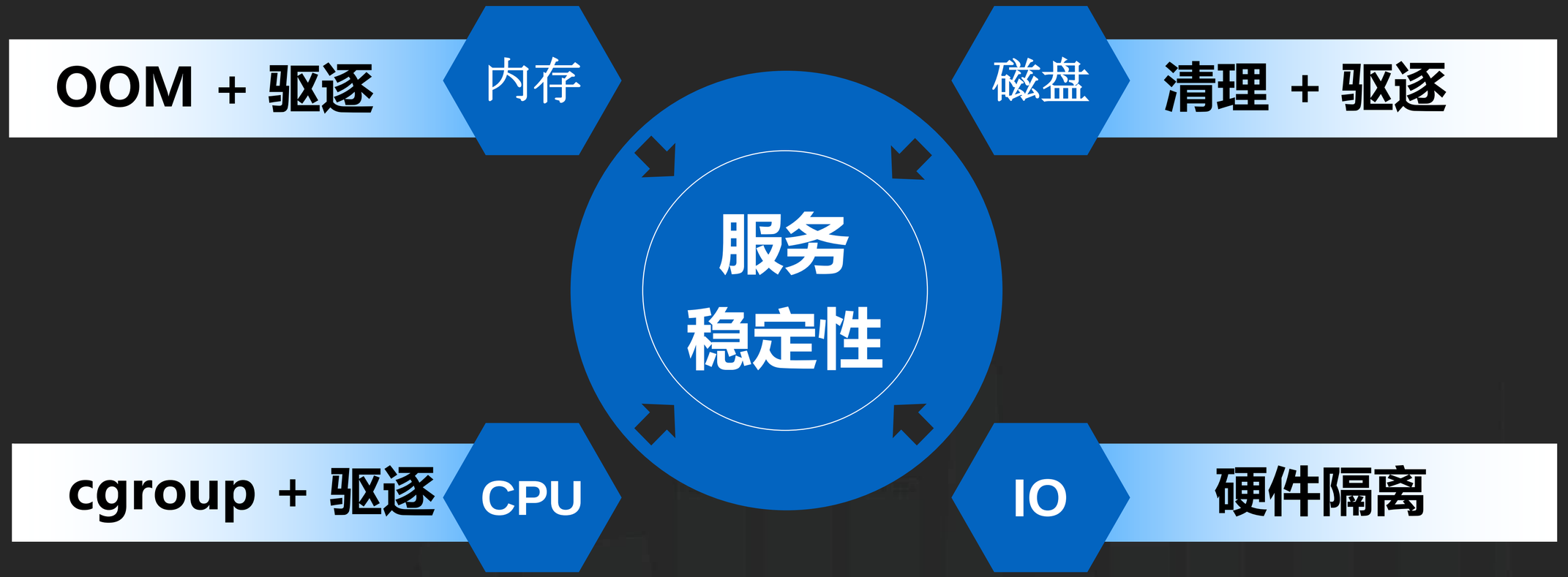
Runc init block

服务稳定性

containerd-shim residue

cgroup, net_cls: iterate the fds of only the tasks which are being migrated

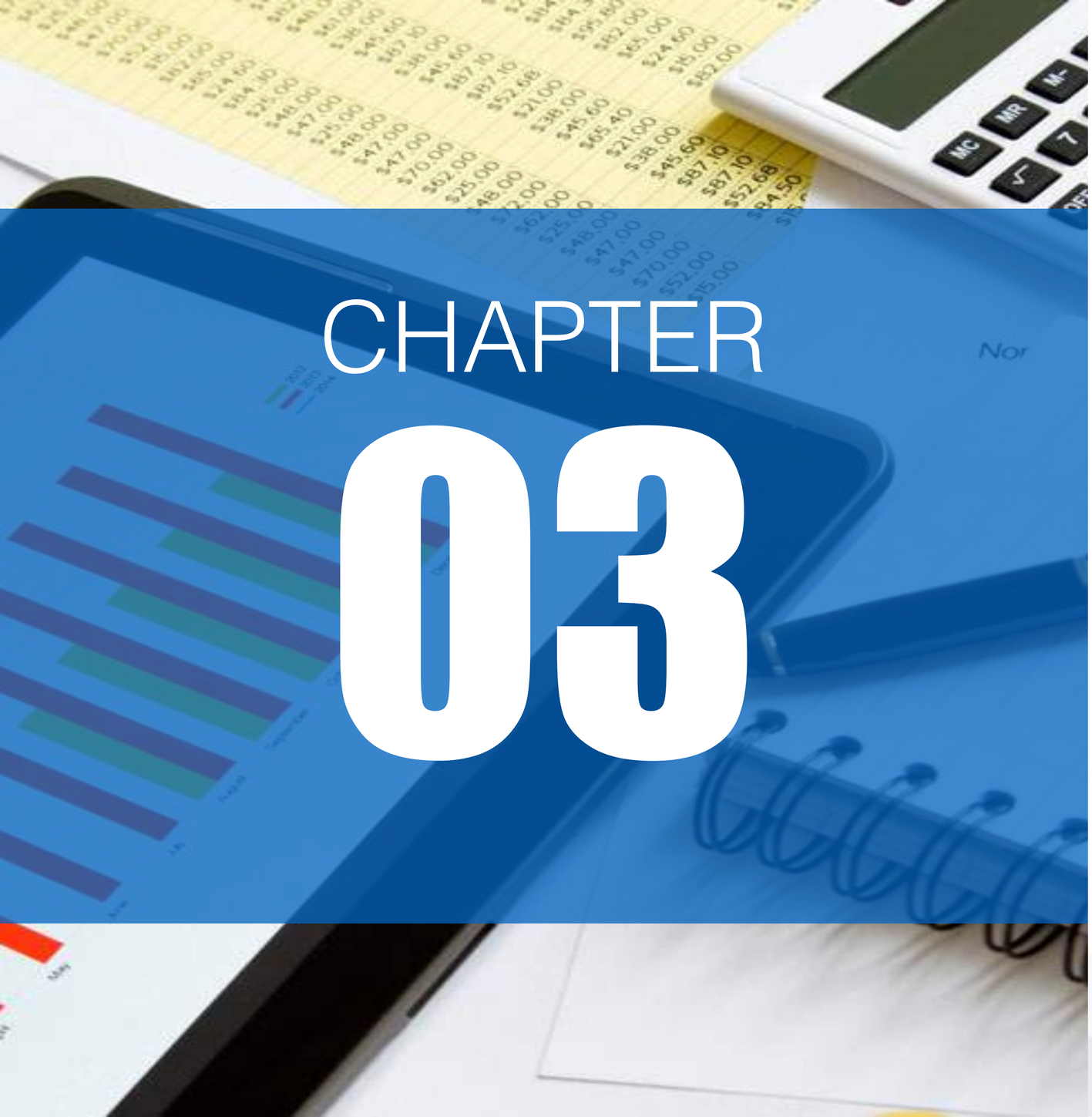
系统层



系统层

USE: For every resource, check utilization, saturation, and errors.

component	type	metric
CPU	utilization	system-wide: `vmstat 1`, "us" + "sy" + "st"; `sar -u`, sum fields except "%idle" and "%iowait"; `dstat -c`, sum fields except "idl" and "wai"; per-cpu: `mpstat -P ALL 1`, sum fields except "%idle" and "%iowait"; `sar -P ALL`, same as `mpstat`; per-process: `top`, "%CPU"; `htop`, "CPU%"; `ps -o pcpu`; `pidstat 1`, "%CPU"; per-kernel-thread: `top`/`htop` ("K" to toggle), where VIRT == 0 (heuristic). [1]
CPU	saturation	system-wide: `vmstat 1`, "r" > CPU count [2]; `sar -q`, "runq-sz" > CPU count; `dstat -p`, "run" > CPU count; per-process: /proc/PID/schedstat 2nd field (sched_info.run_delay); `perf sched latency` (shows "Average" and "Maximum" delay per-schedule); dynamic tracing, eg, SystemTap schedtimes.stp "queued(us)" [3]
CPU	errors	`perf` (LPE) if processor specific error events (CPC) are available; eg, AMD64's "04Ah Single-bit ECC Errors Recorded by Scrubber" [4]
Memory capacity	utilization	system-wide: `free -m`, "Mem:" (main memory), "Swap:" (virtual memory); `vmstat 1`, "free" (main memory), "swap" (virtual memory); `sar -r`, "%memused"; `dstat -m`, "free"; `slabtop -s c` for kmem slab usage; per-process: `top`/`htop`, "RES" (resident main memory), "VIRT" (virtual memory), "Mem" for system-wide summary
Memory capacity	saturation	system-wide: `vmstat 1`, "si"/"so" (swapping); `sar -B`, "pgscank" + "pgscand" (scanning); `sar -W`; per-process: 10th field (min_flt) from /proc/PID/stat for minor-fault rate, or dynamic tracing [5]; OOM killer: `dmesg | grep killed`
Memory capacity	errors	`dmesg` for physical failures; dynamic tracing, eg, SystemTap uprobes for failed malloc()s
Network Interfaces	utilization	`sar -n DEV 1`, "rxKB/s"/max "txKB/s"/max; `ip -s link`, RX/TX tput / max bandwidth; /proc/net/dev, "bytes" RX/TX tput/max; nicstat "%Util" [6]
Network Interfaces	saturation	`ifconfig`, "overruns", "dropped"; `netstat -s`, "segments retransmited"; `sar -n EDEV`, *drop and *fifo metrics; /proc/net/dev, RX/TX "drop"; nicstat "Sat" [6]; dynamic tracing for other TCP/IP stack queueing [7]
Network Interfaces	errors	`ifconfig`, "errors", "dropped"; `netstat -i`, "RX-ERR"/"TX-ERR"; `ip -s link`, "errors"; `sar -n EDEV`, "rxerr/s" "txerr/s"; /proc/net/dev, "errs", "drop"; extra counters may be under /sys/class/net/...; dynamic tracing of driver function returns 76]
Storage device I/O	utilization	system-wide: `iostat -xz 1`, "%util"; `sar -d`, "%util"; per-process: iotop; `pidstat -d`; /proc/PID/sched "se.statistics.iowait_sum"
Storage device I/O	saturation	`iostat -xnz 1`, "avgqu-sz" > 1, or high "await"; `sar -d` same; LPE block probes for queue length/latency; dynamic/static tracing of I/O subsystem (incl. LPE block probes)
Storage device I/O	errors	/sys/devices/.../ioerr_cnt; `smartctl`; dynamic/static tracing of I/O subsystem response codes [8]
Storage capacity	utilization	swap: `swapon -s`; `free`; /proc/meminfo "SwapFree"/"SwapTotal"; file systems: "df -h"
Storage capacity	saturation	not sure this one makes sense - once it's full, ENOSPC
Storage capacity	errors	`strace` for ENOSPC; dynamic tracing for ENOSPC; /var/log/messages errs, depending on FS



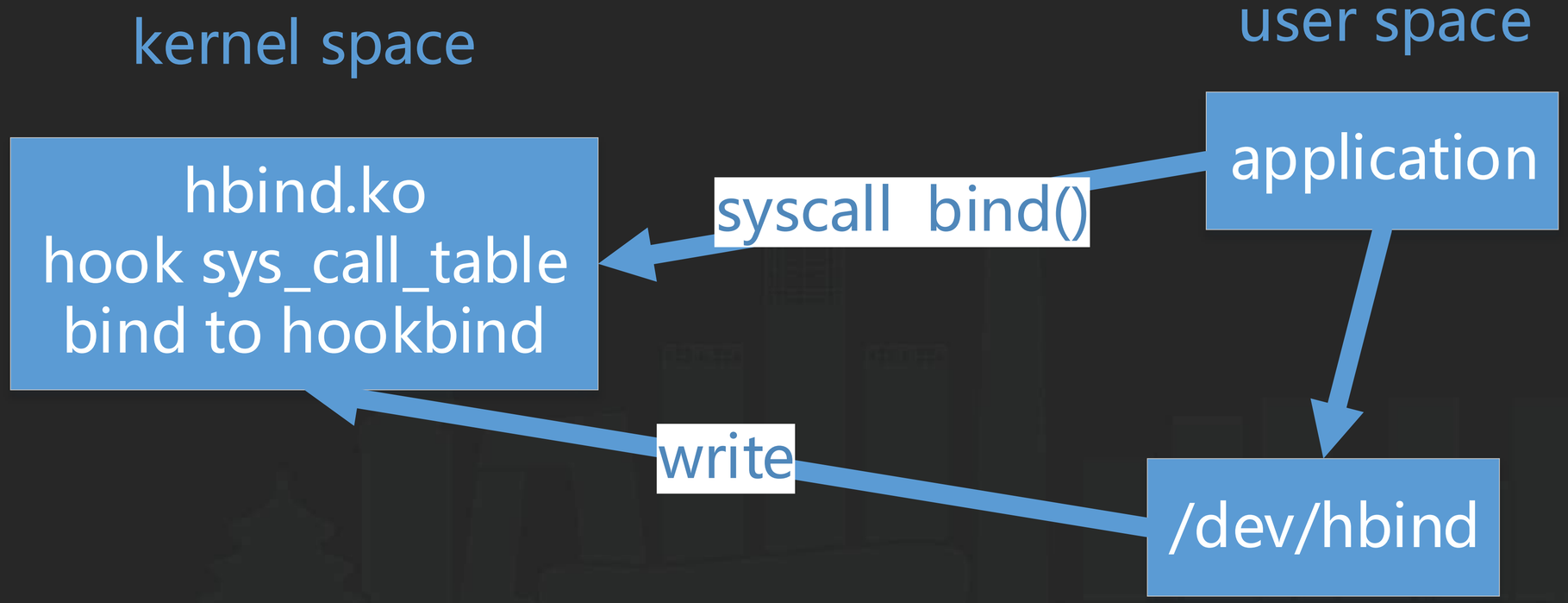
CHAPTER

03

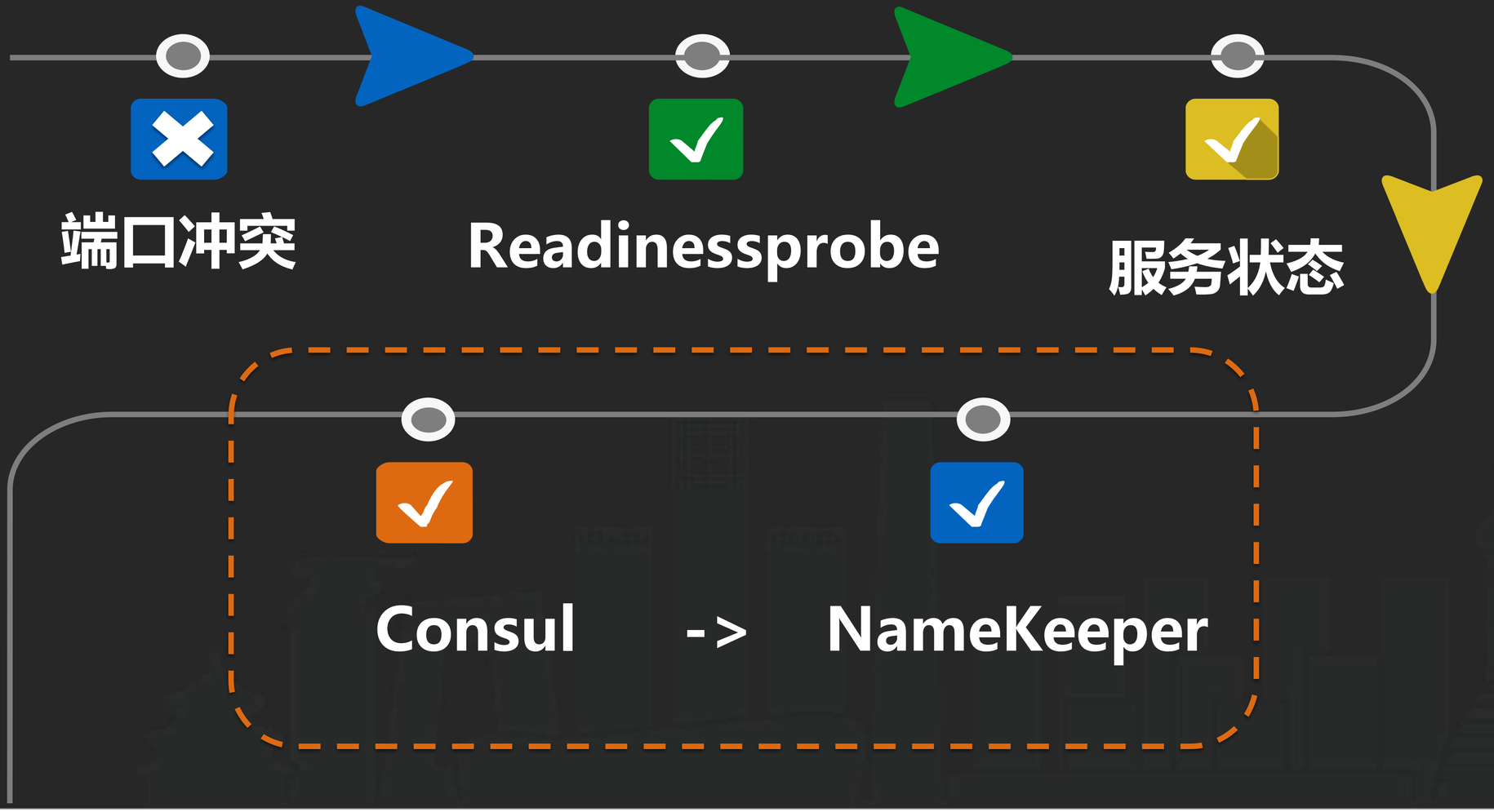
网络

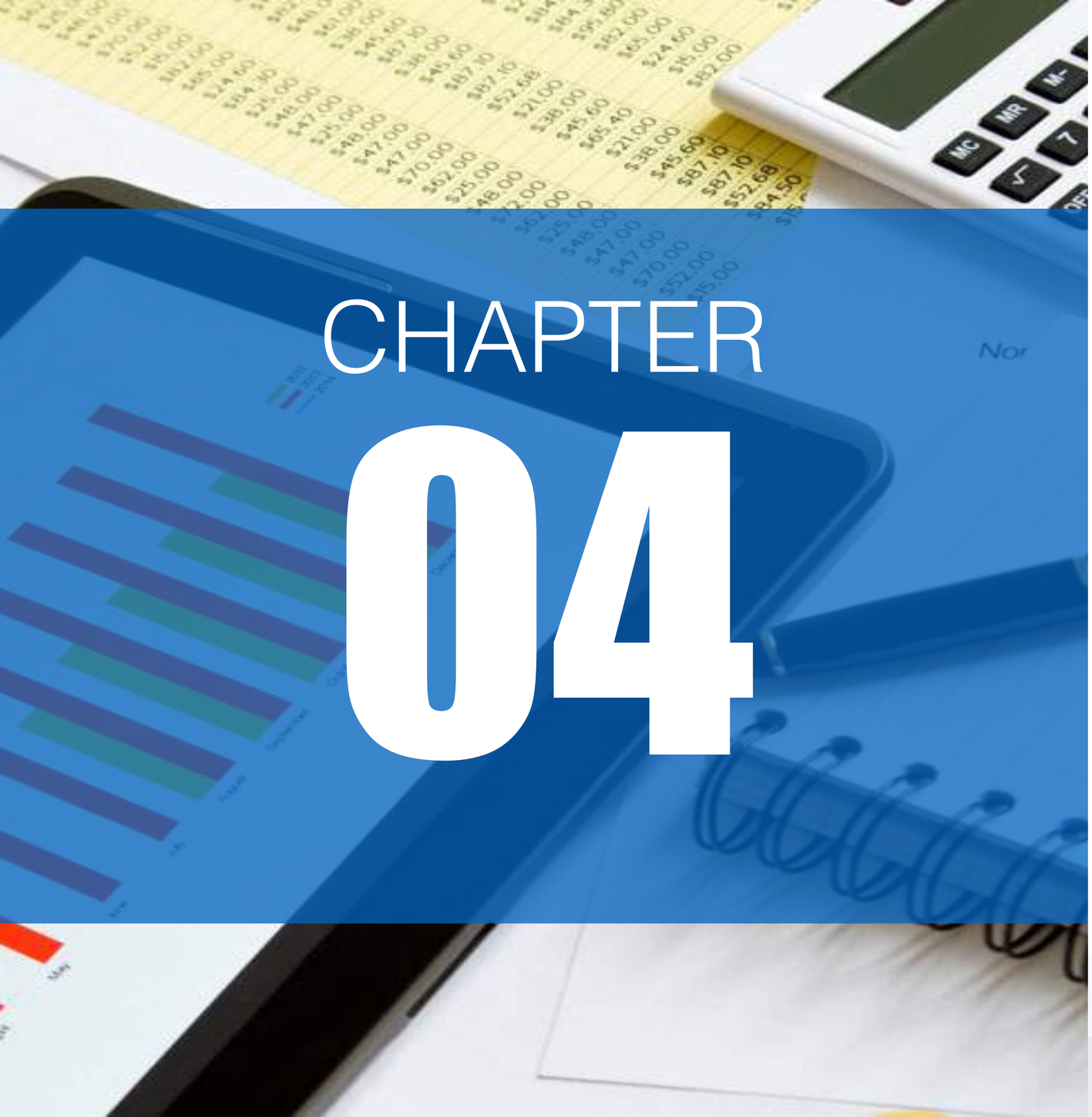
网络模式

Bridge **Host** **AutoHost**



服务发现





CHAPTER

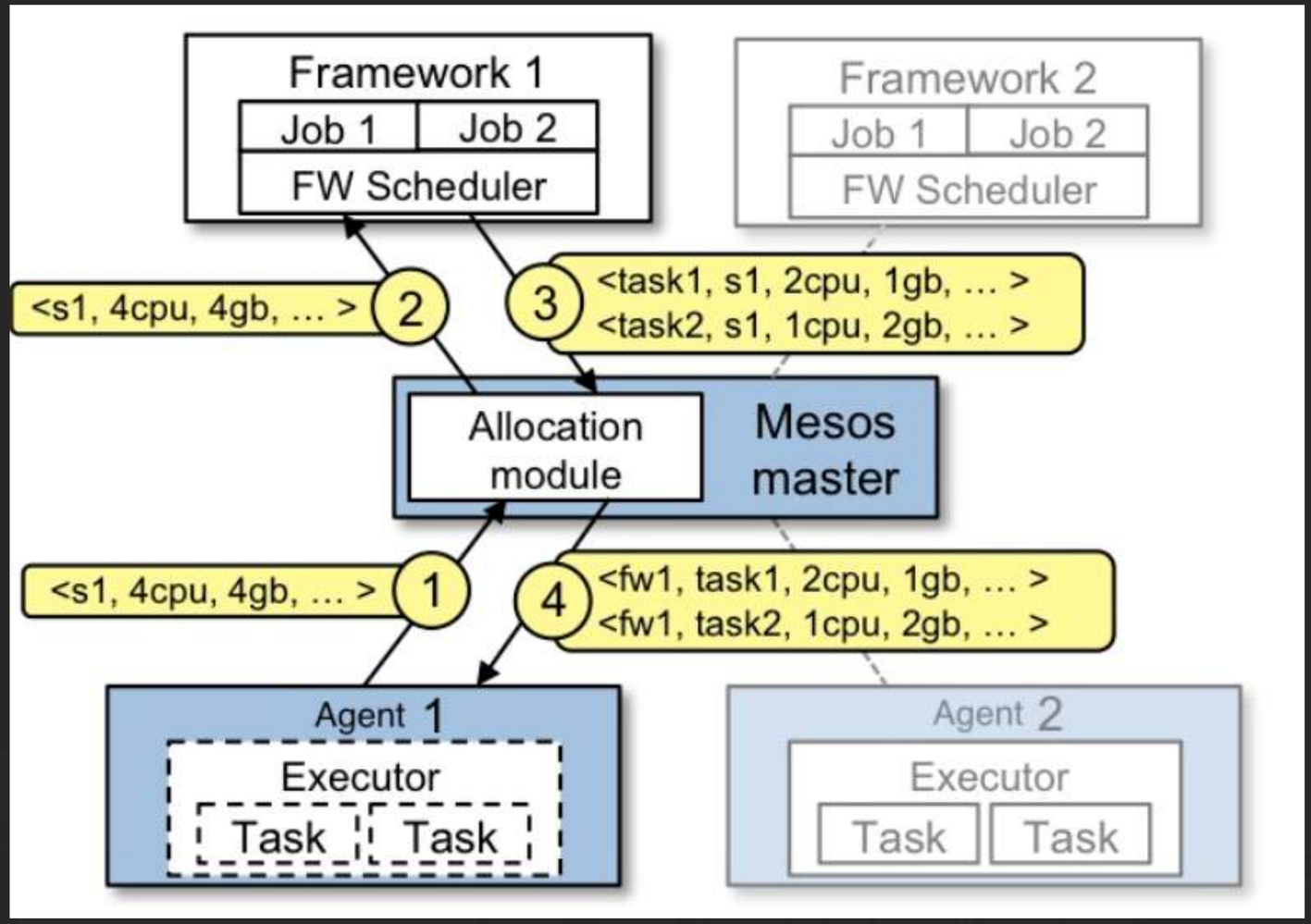
04

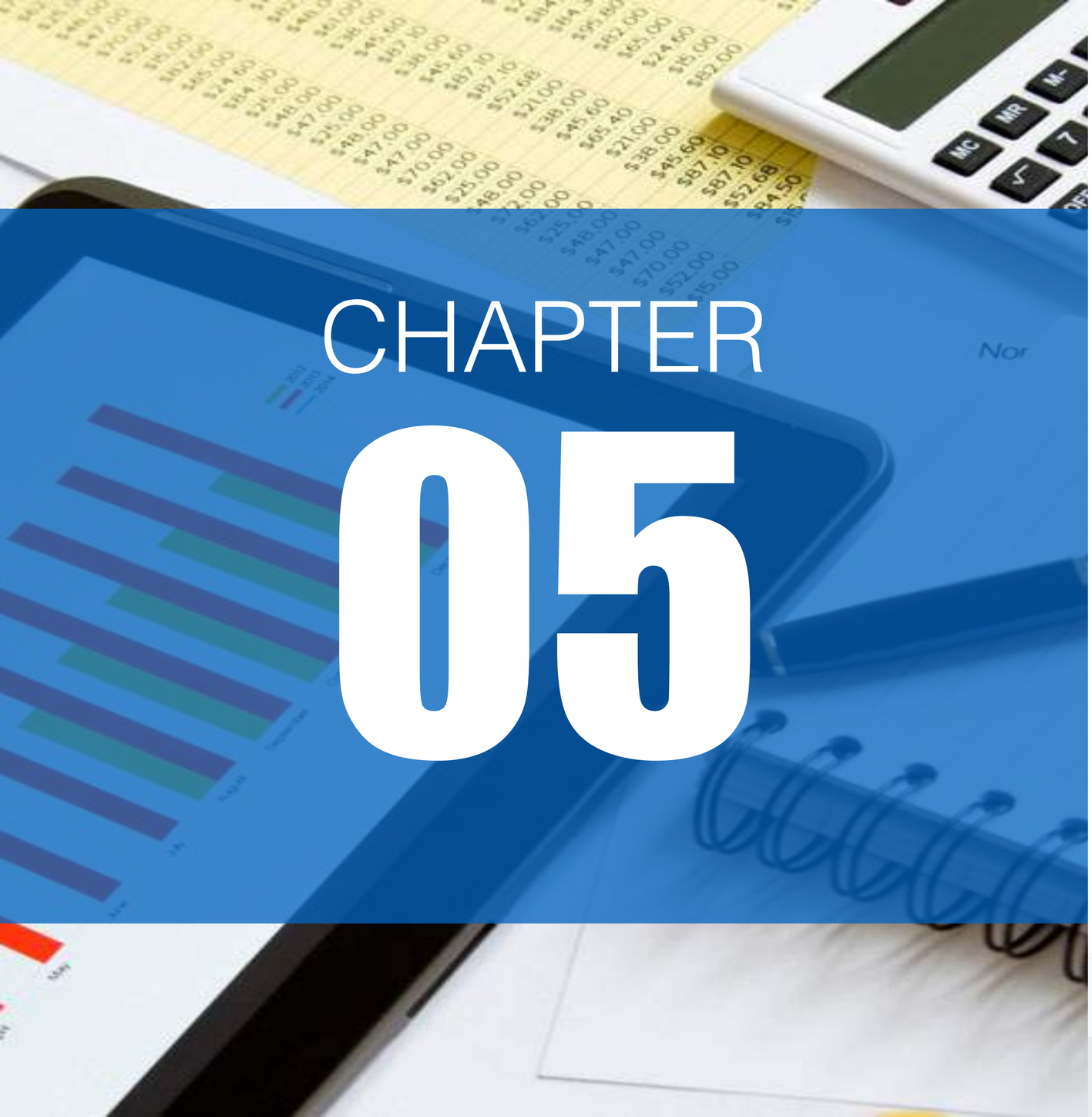
物理机管理

物理机管理

Agent -> DaemonSet

Mesos (TCE Allocator)





CHAPTER

05

收益


1
资源利用率
2
运维效率
3
流量调度与业务解耦