阿里云CDN技术演进之路

阿里云-核心系统部-Web平台
朱照远（叔度）
2014-12-19

大纲

- 简介
- 历年技术架构
- 现有架构和实现

什么是CDN

- 内容分发网络（Content Delivery Network）
- 在不同的地点缓存内容，通过负载均衡等技术将用户请求定向到最合适的缓存服务器上获取内容，加速用户对网站的访问速度

阿里云CDN的特点和优势

- 稳定
  - 节点资源丰富，全球260+节点、7Tbps带宽处理能力
  - 技术领先，自主开发的缓存、调度、安全、业务管理等系统
- 安全
  - 全网部署安全防护模块，1.6Tbps的DDoS防护能力
  - 基于大数据分析，快速准确识别攻击，实时阻断
- 易用
  - 自助化业务部署，全程无需人工审核
  - 可通过Open API管理
- 低成本
  - 按需计费，根据实际使用流量后付费

客户评价

- “阿里云CDN除提供卓越的速度体验外，同时提供企业级的安全防护，为我们的服务保驾护航。7*24小时的售后工程师快速响应，帮助业务健康持续的增长。”——崩坏学园
- “阿里云CDN按量付费这个特点，让我们避免像传统视频业务那样一次性投入高额成本。同时OSS+CDN的方案完美解决我们海量内容存储和分发的业务场景。”——趣拍
- “阿里云OSS与CDN的无缝连接为CDN的卓越表现奠定了基础，稳定可靠的产品性能保障了唱吧数据的快速可靠访问。完善的售后服务也保证了问题的快速解决。”——唱吧

阿里云
aliyun.com

历年技术架构

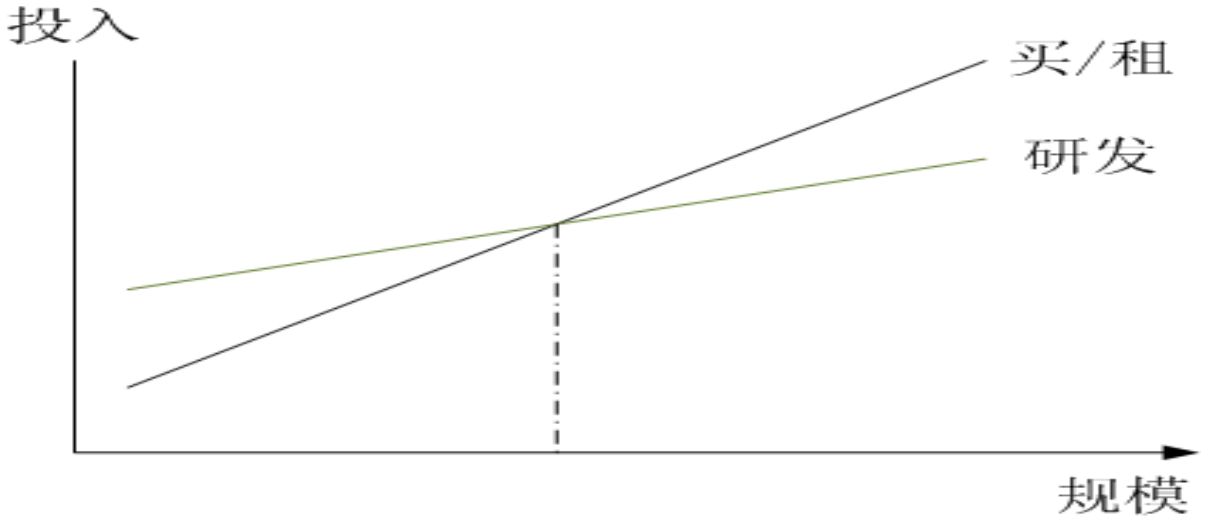
阿里云CDN的前世今生

- 2008年~2011年，淘宝CDN
  - 由淘宝技术部发起
  - 为淘宝网提供服务
- 2011年~2014年2月，阿里CDN
  - 淘宝CDN发展成阿里CDN
  - 为阿里巴巴集团所有子公司提供服务
- 2014年2月~，阿里云CDN
  - 阿里CDN发展成阿里云CDN
  - 为阿里巴巴集团所有子公司提供服务
  - 同时将自身的资源、技术以云计算的方式输出，对外服务

2009年之前

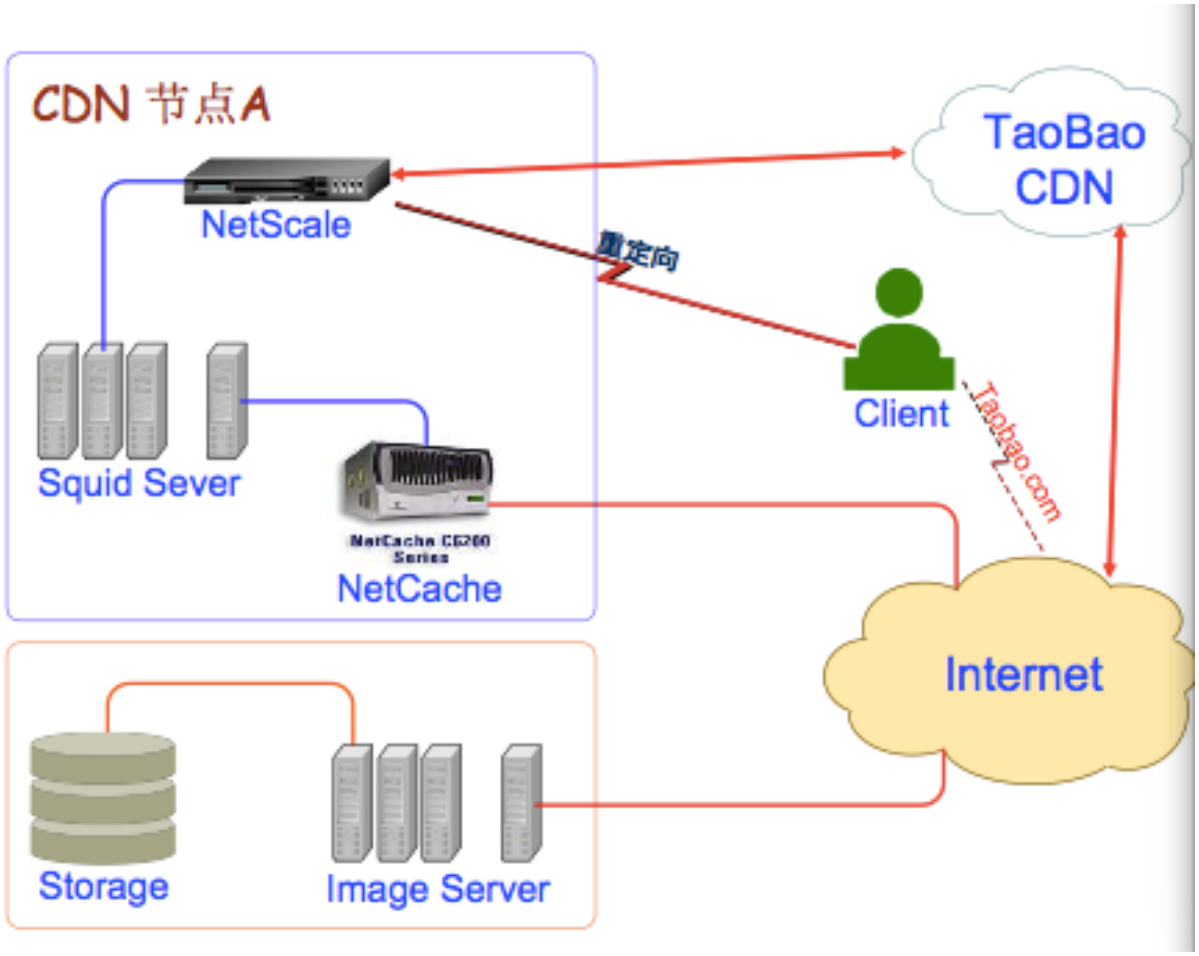
阿里云 aliyun.com

- 2008年之前使用商业CDN提供的服务
- 2008年淘宝技术团队开始自建CDN
  - 商用软件不能满足大规模系统的需求
  - 采用开源软件与自主开发相结合，有更好的可控性和更大的优化空间，系统上有更高的可扩展性
  - 规模效应，研发投入都是值得的



2009年之前技术架构

- 关键组件
  - 3DNS
  - NetScaler
  - Squid
  - NetCache

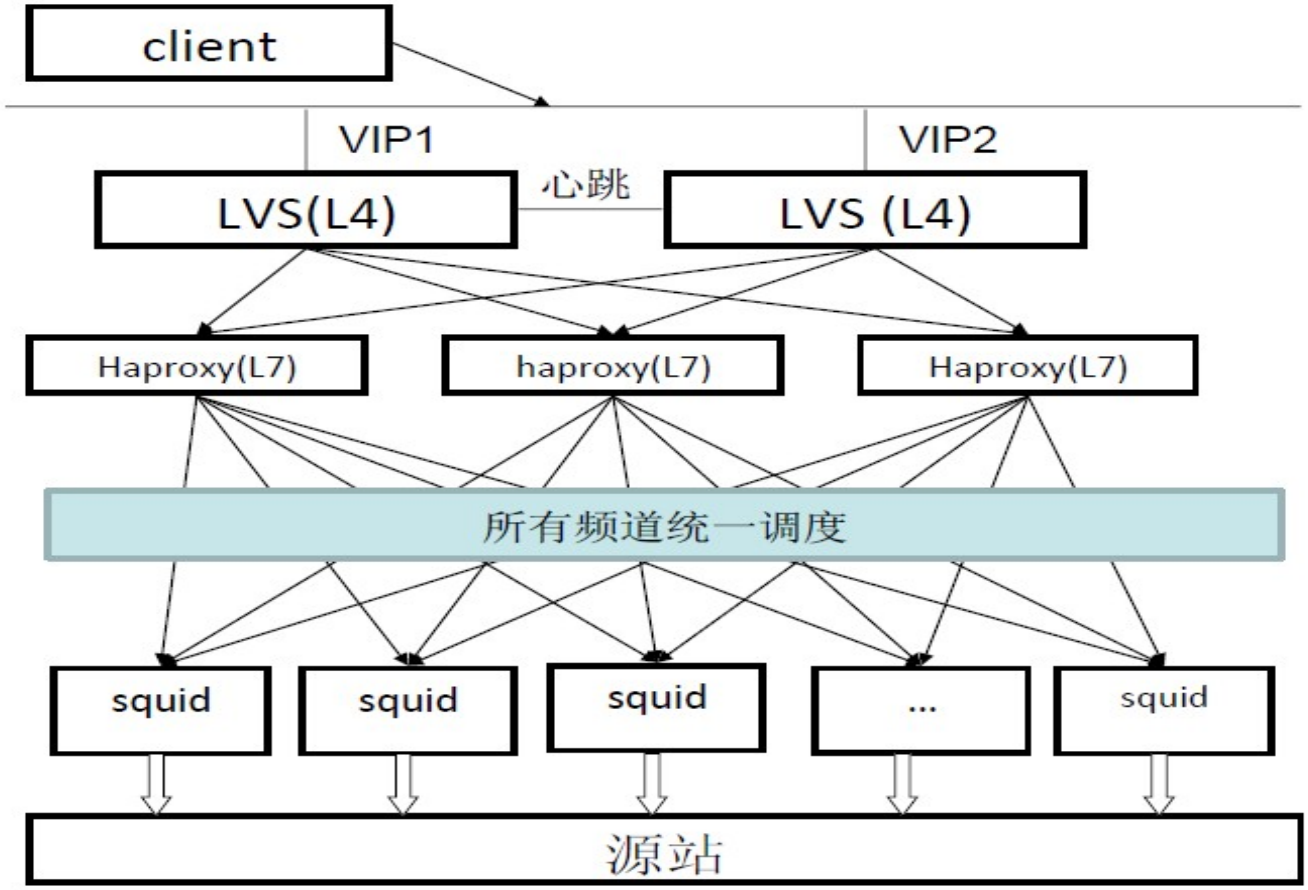


2009年~2011年

- 规模
  - 2008-2009年67Gbps
    - 14个节点
  - 2010年318Gbps
    - 43个节点
    - 单节点容量10G
  - 2011年861Gbps
    - 103个节点
    - 单节点容量10G

2009年~2011年的技术架构

- 关键组件
  - GTM
  - LVS
  - HAProxy
  - Squid

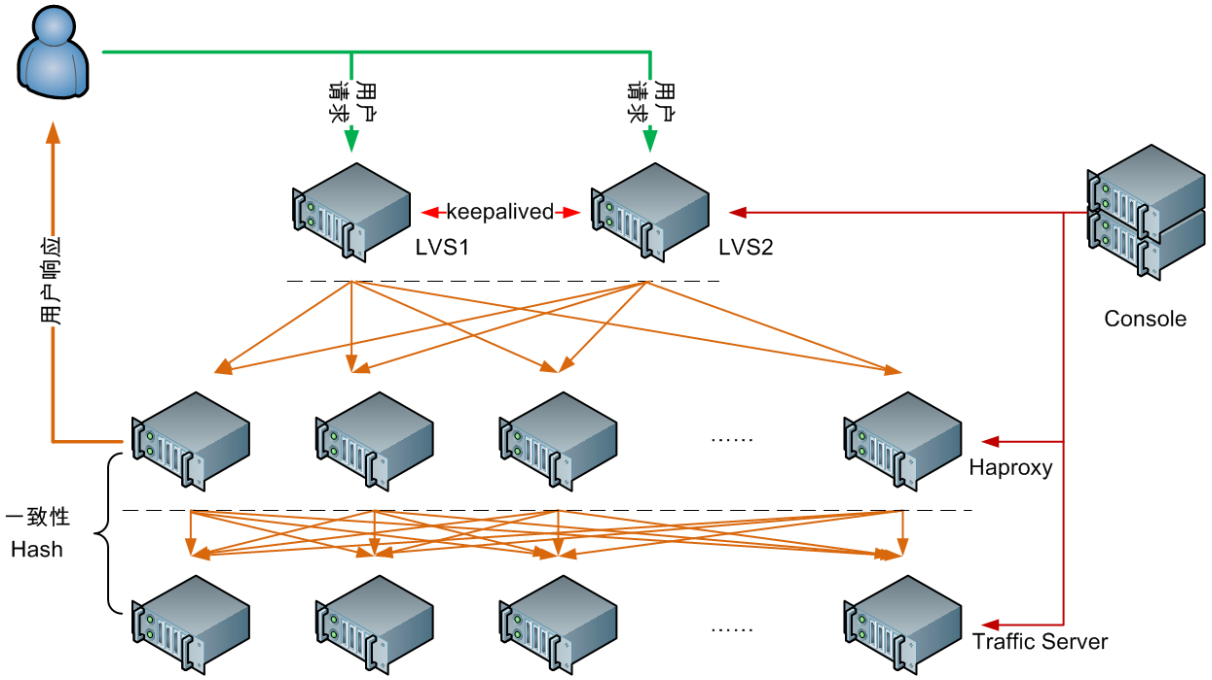


2012年~2013年

- 规模
  - 2012年2100Gbps
    - 130个节点
    - 单节点容量20-30G
  - 2013年3400Gbps
    - 230个节点
    - 单节点容量40G

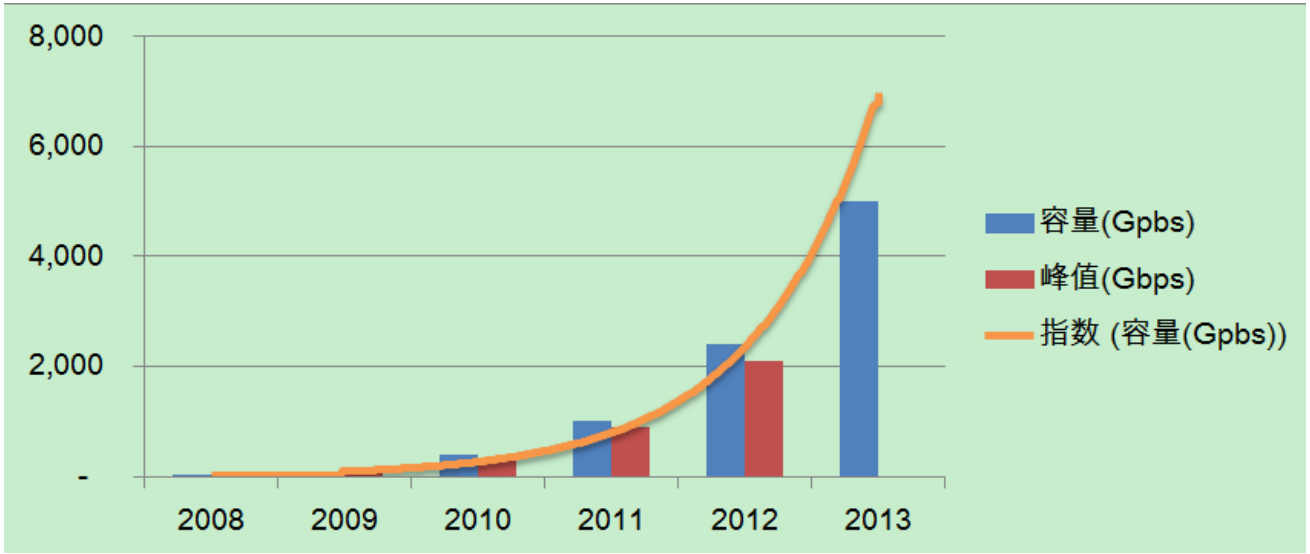
2012年~2013年的技术架构

- 关键组件
  - Pharos
  - LVS
  - HAProxy
  - Traffic Server
  - Tengine
  - Swift



2014年~

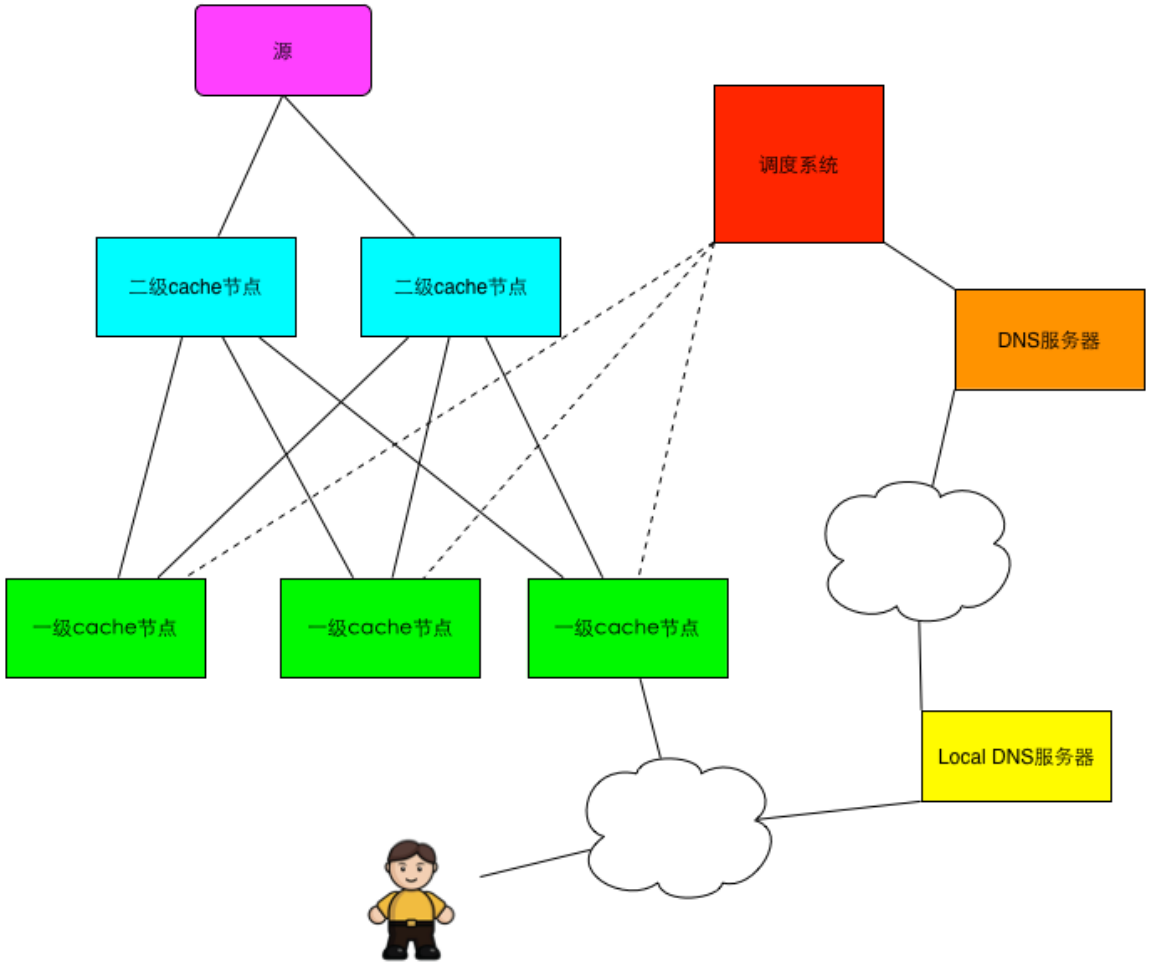
- 全球30个国家260+个节点
- 7Tbps服务能力储备
- 1机柜单节点40Gbps服务能力
- 处于业界前沿的开源技术研究及开发
- 阿里云CDN于2014年2月正式对外提供服务



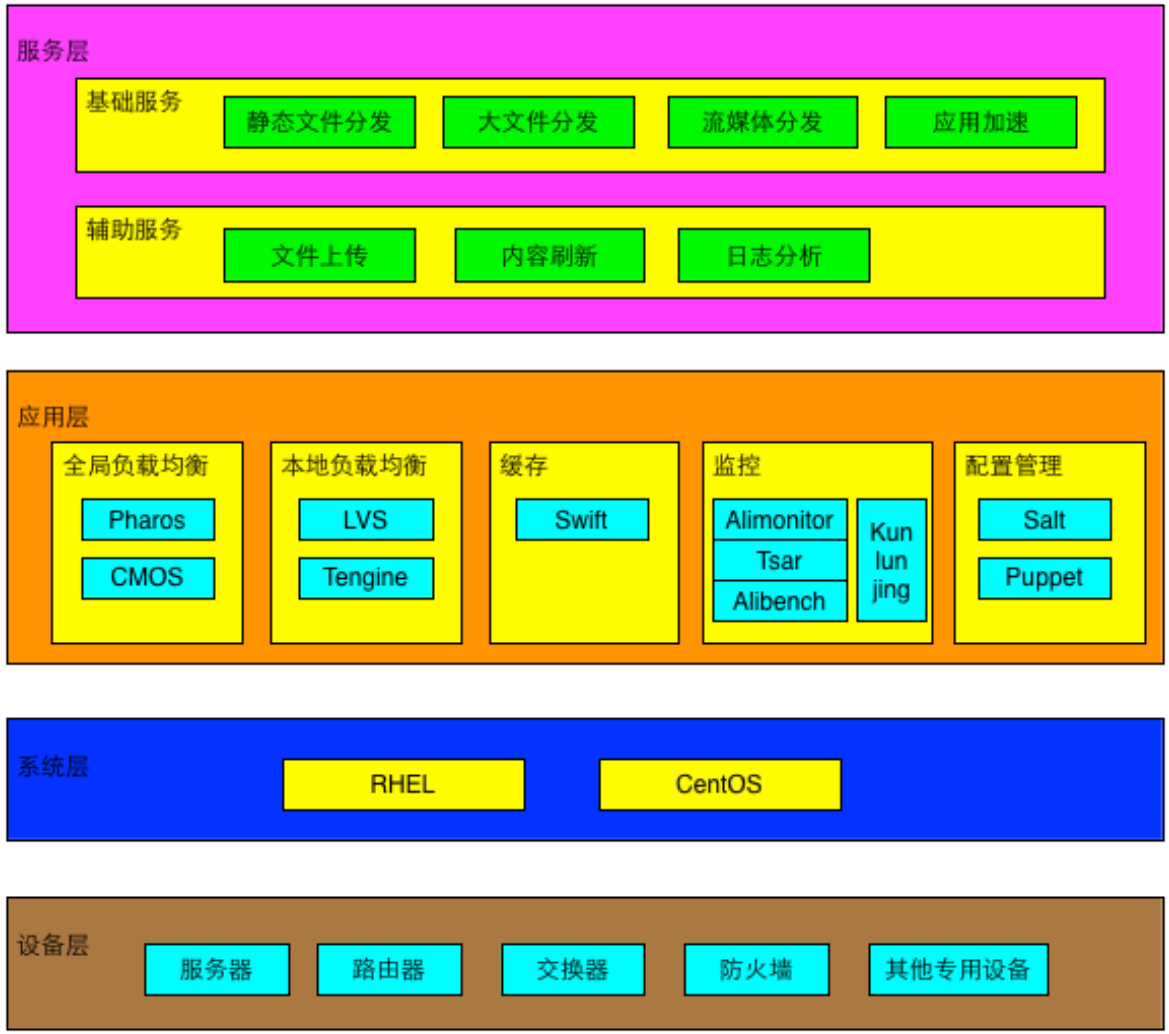
阿里云
aliyun.com

现有架构和实现

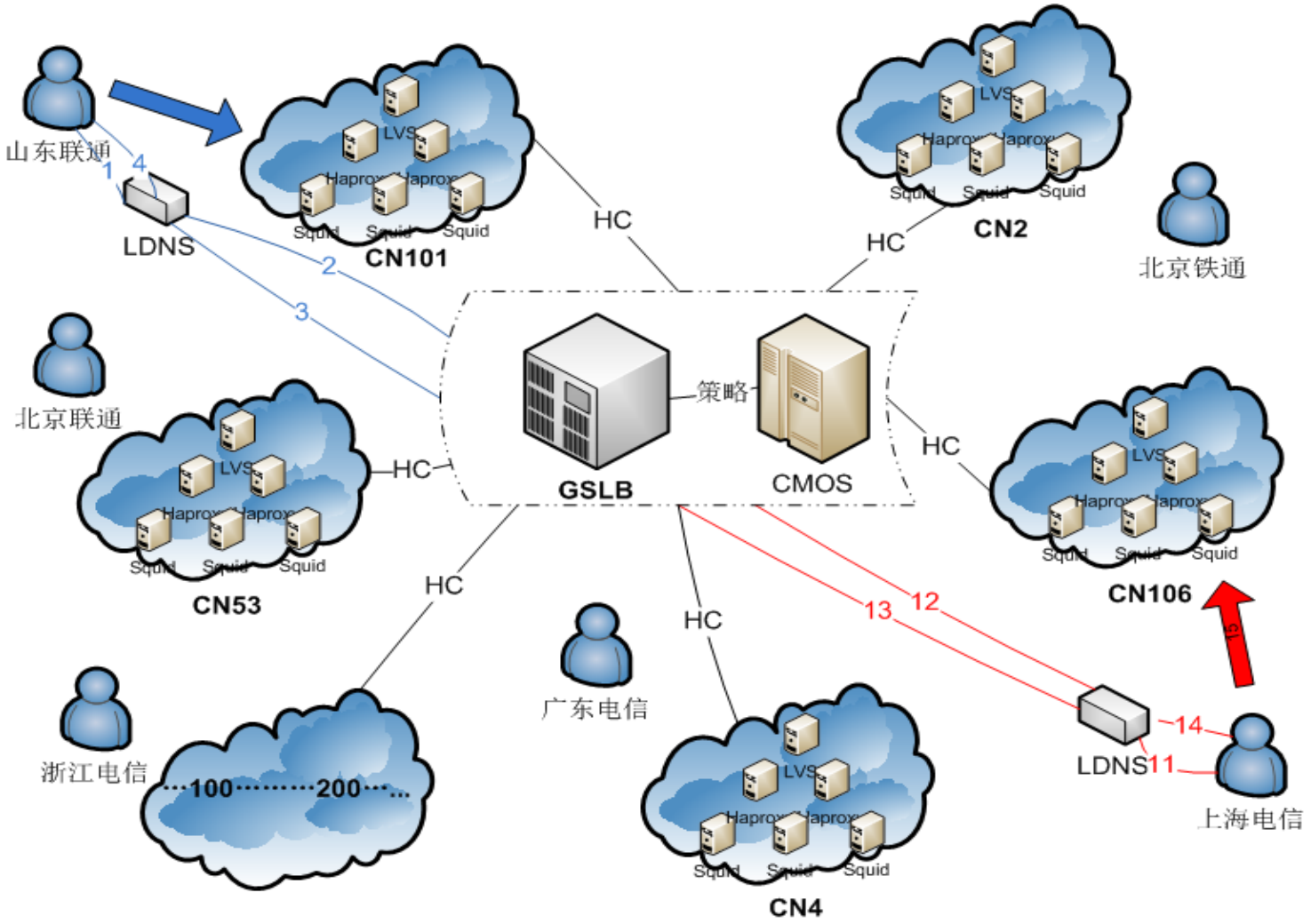
阿里云CDN大图


源
调度系统
二级cache节点
二级cache节点
DNS服务器
一级cache节点
一级cache节点
一级cache节点
Local DNS服务器

组件分层


服务层
基础服务
静态文件分发
大文件分发
流媒体分发
应用加速
辅助服务
文件上传
内容刷新
日志分析
应用层
全局负载均衡
Pharos
CMOS
本地负载均衡
LVS
Tengine
缓存
Swift
监控
Alimonitor
Tsar
Alibench
Kun lun jing
配置管理
Salt
Puppet
系统层
RHEL
CentOS
设备层
服务器
路由器
交换器
防火墙
其他专用设备

阿里云CDN的大脑：全局流量调度



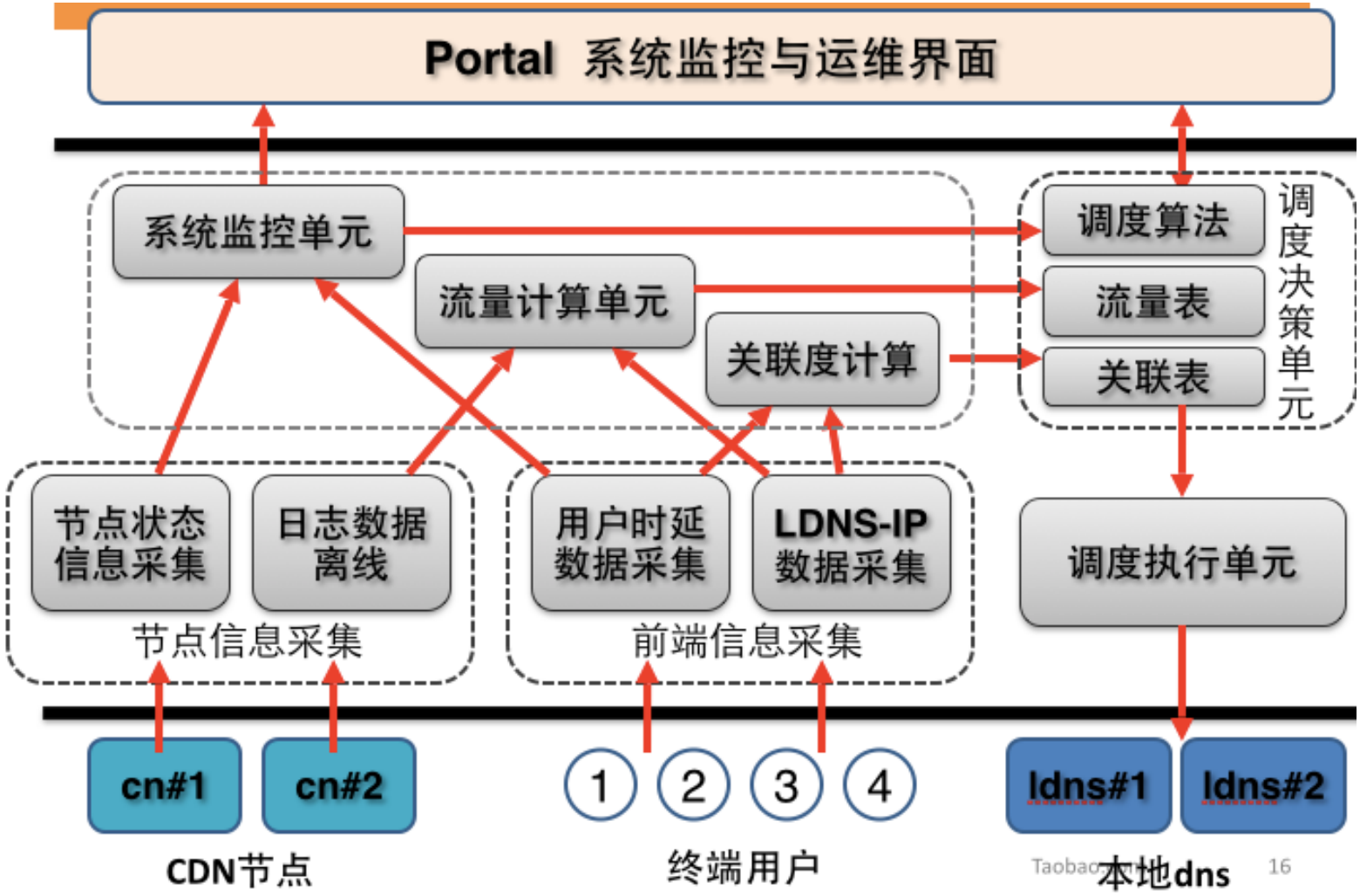
DNS服务器：Pharos

- 自主研发的调度系统，可控性，协议扩展性都更好，也省下了采购商用设备的成本
- 单机高性能，支持百万级别的域名
- 支持多级的策略调度，节点故障不会造成用户的不可用
- 支持EDNS扩展协议
- 多系统联动，与安全防御系统，刷新系统，内容管理系统联动
- Portal，API，tcheck等多种管理方式

实时调度系统：CMOS

- 数据化的调度
- 流量完全可控，降低了抖动造成的带宽成本
- LDNS级别、节点级别的流量预测，流量峰值到来前提前应对
- 精确、准实时的流量调度
- 平均误差小于15%，精度可以到5M级别
- 单个Local DNS级别的调度
- 5分钟级别的准实时
- 调度质量、准确度的提升，直接影响着用户体验
- 自动化的调度
- 只要描述调度的场景，设定约束条件，自动计算，生成适应的策略，更新pharos

Pharos+CMOS架构



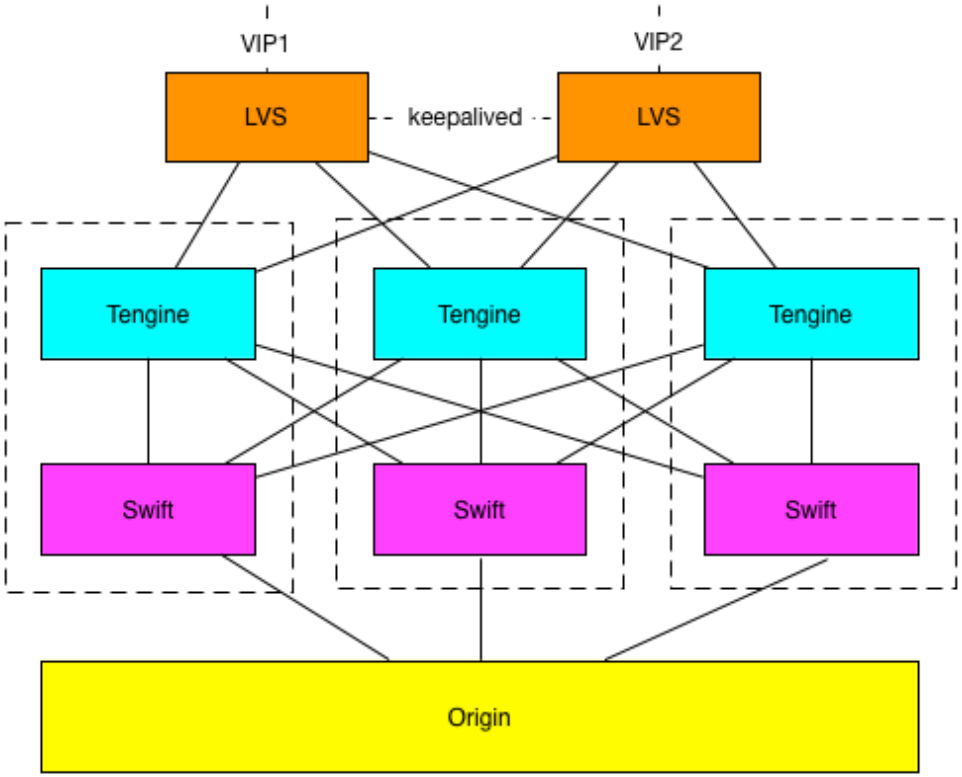
调度准确性的重要基础：IP地址库

- 数据采集，多个数据源
- 数据运算与评估（加权投票、评估体系）
  - 对各个数据源的数据质量，设置不同权重，进行投票
  - 权重的设置，是根据数据源质量的评估结果进行设置，质量高，权重高，否则相反
  - 根据淘宝包裹地址和IP做数据校验
  - 根据上次的结果进行迭代

	覆盖度（粗）	覆盖度（细）	准确度	有效比
国家	100%	100%	100%	100%
省/直辖市/自治区	99.92%	99.98%	99.89%	99.87%
市	93.61%	99.75%	96.52%	96.28%

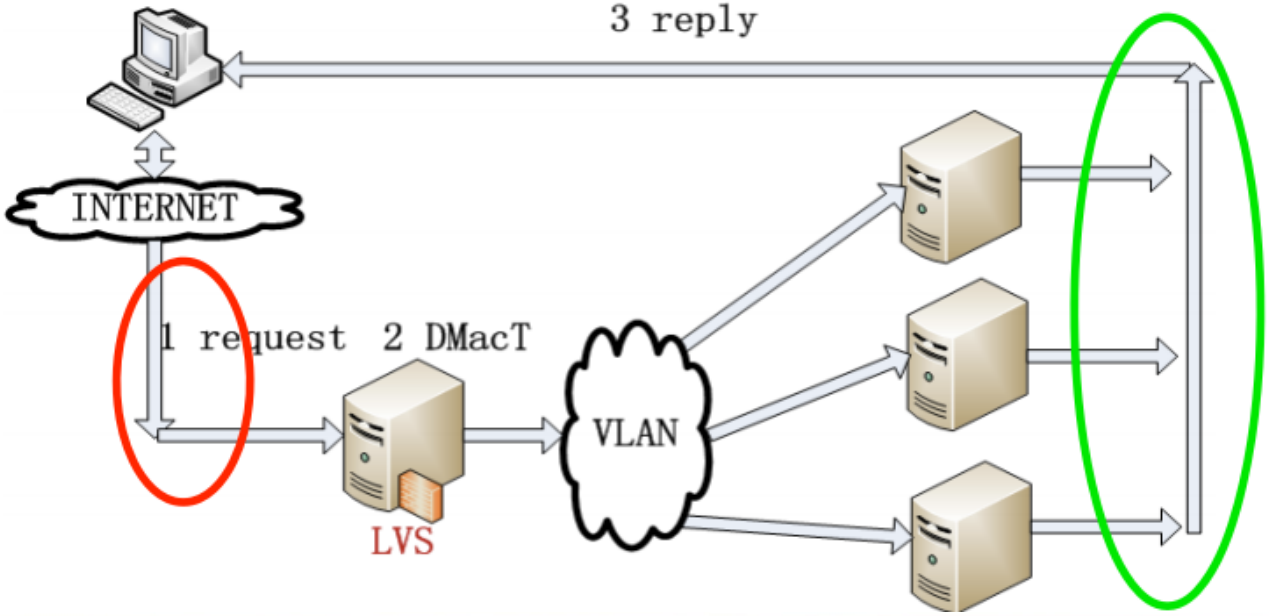
阿里CDN节点系统：内部架构图

- 关键组件
  - LVS做四层负载均衡
  - Tengine做七层负载均衡
    - 安全
    - 业务逻辑处理
  - Swift做HTTP缓存
    - 高性能cache
    - 磁盘（SSD/SATA）



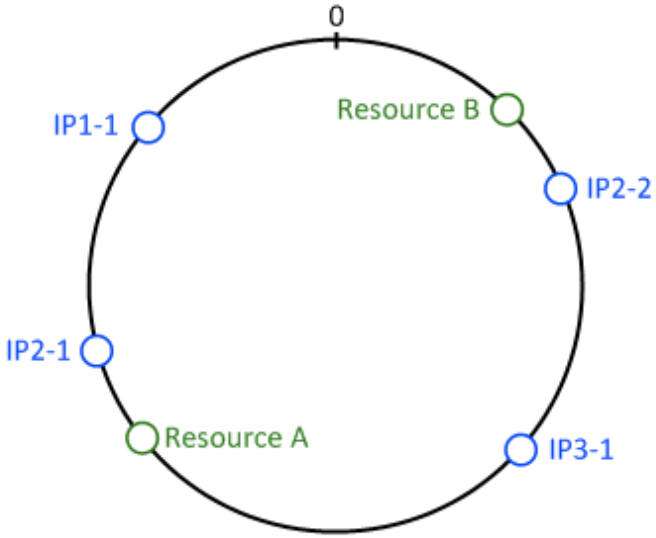
四层负载均衡：LVS

- DR模式
  - IN的流量经过LVS，OUT的不经过
- 负载均衡算法采用wrr
- 双LVS做Active-Active互备，中间有心跳监测



七层负载均衡：Tengine

- 阿里基于Nginx开发的高性能HTTP服务器
  - 已经开源于：http://tengine.taobao.org
- 一致性hash（consistent hashing）
  - 提高命中率
  - 降低抖动
- 主动健康检查
- SPDY v3支持
- SO_REUSEPORT支持
  - 提高worker进程之间的均衡性
  - 降低CPU使用
- 热点对象发现

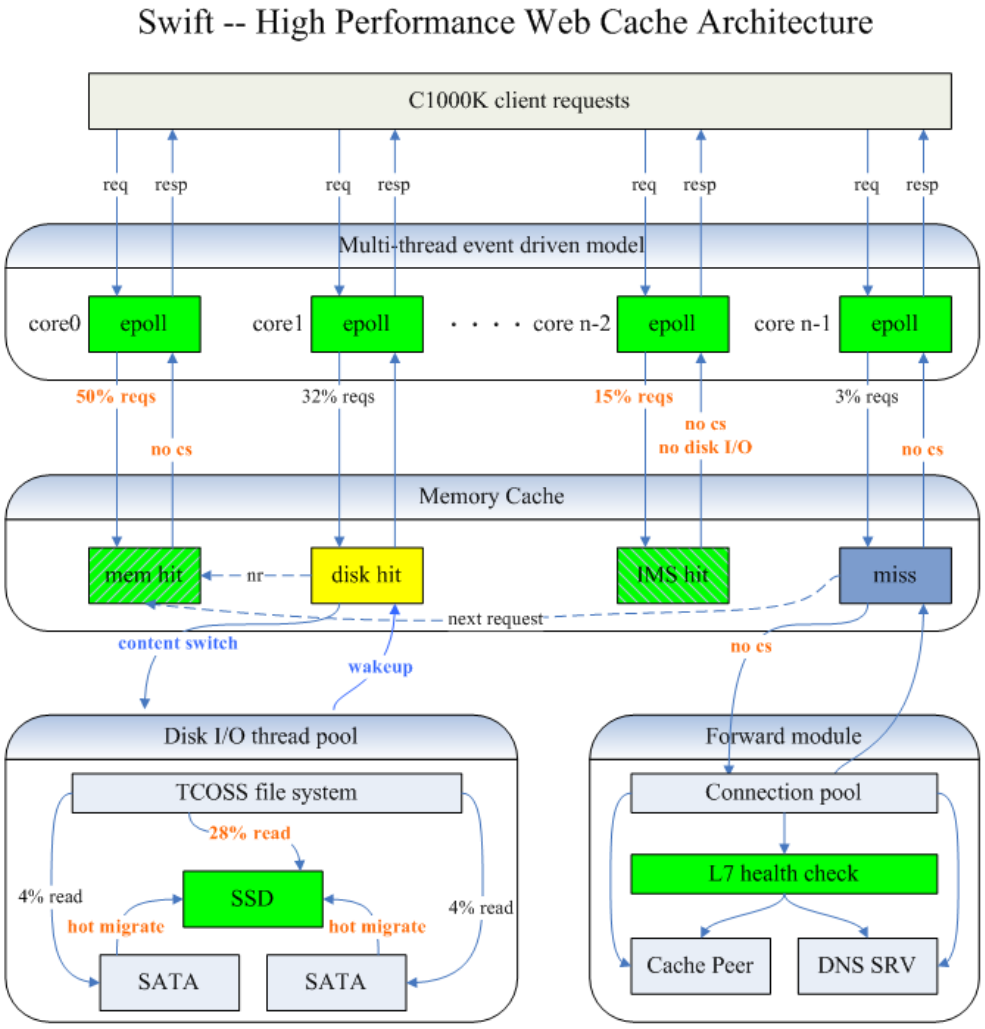


阿里HTTP缓存服务器：Swift

- 基础功能
  - HTTP/1.1协议、proxy功能
  - 内存缓存、磁盘存储
  - HTTPS协议关键特性的支持
- 业务功能
  - 精确purge/dir purge/正则purge
  - 鉴权X-Referer-Acl
  - ESI+gzip
- 运维和配置相关功能
  - 按照域名配置的功能
  - if、变量支持
  - 磁盘容错。磁盘为只读不再进行写操作；磁盘不可读将磁盘摘掉
  - 丰富的统计信息

Swift总体架构图

- HTTP处理引擎
- 回源
- 存储
- 索引
- 内容管理子系统



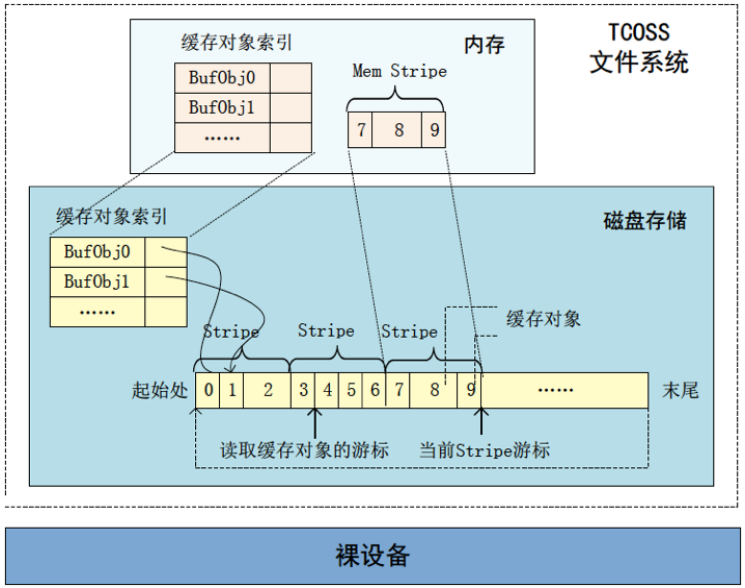
Swift性能优化点

阿里云 aliyun.com

- 多线程事件驱动网络模型
- 减小线程间上下文切换
- 内存命中，一个请求只需要一个线程来处理
- 消除在万兆网卡上网络处理的瓶颈
- 304的请求没有Disk I/O
- 使用trie树实现快速匹配，减少ACL字符串匹配
- 使用完美hash计算header id，实现批量拷贝、删除响应头
- 使用libaio（Linux内核AIO）优化IO操作
- 大文件分片不同片可以分到所有的磁盘上，可以按片做热点
- 七层负载均衡、热点cache
- 分级存储和热点迁移

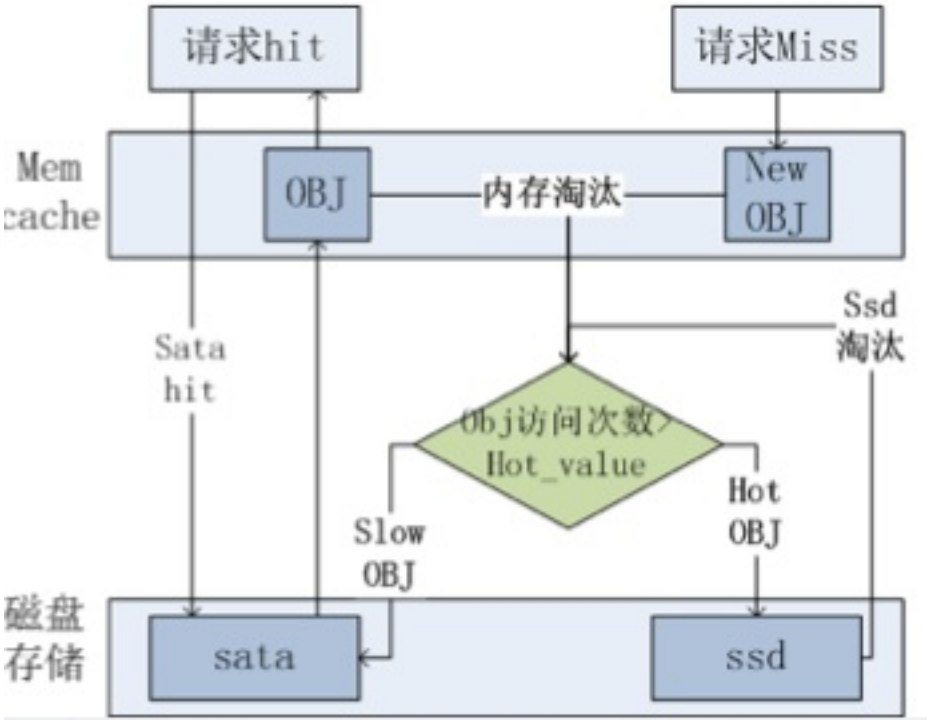
Swift的文件存储系统

- TCOSS（Taobao Cyclic Object Storage System）
  - 基于Squid的COSS系统做的定制开发
  - 支持裸盘热拔插
  - COSS对象访问导致平均2.13次I/O访问
  - TCOSS对象访问导致平均1次IO访



Swift热点迁移算法

- 三层存储
  - 内存
  - SSD
  - SATA
- 根据对象热度决定到哪层



Tengine+Swift性能优化

- 集群的大文件分片缓存功能
- 基于HTTP分段压缩算法
- 利用SPDY的多路复用技术
  - 减少三路握手和慢启动的影响
  - 减少对本地端口的占用

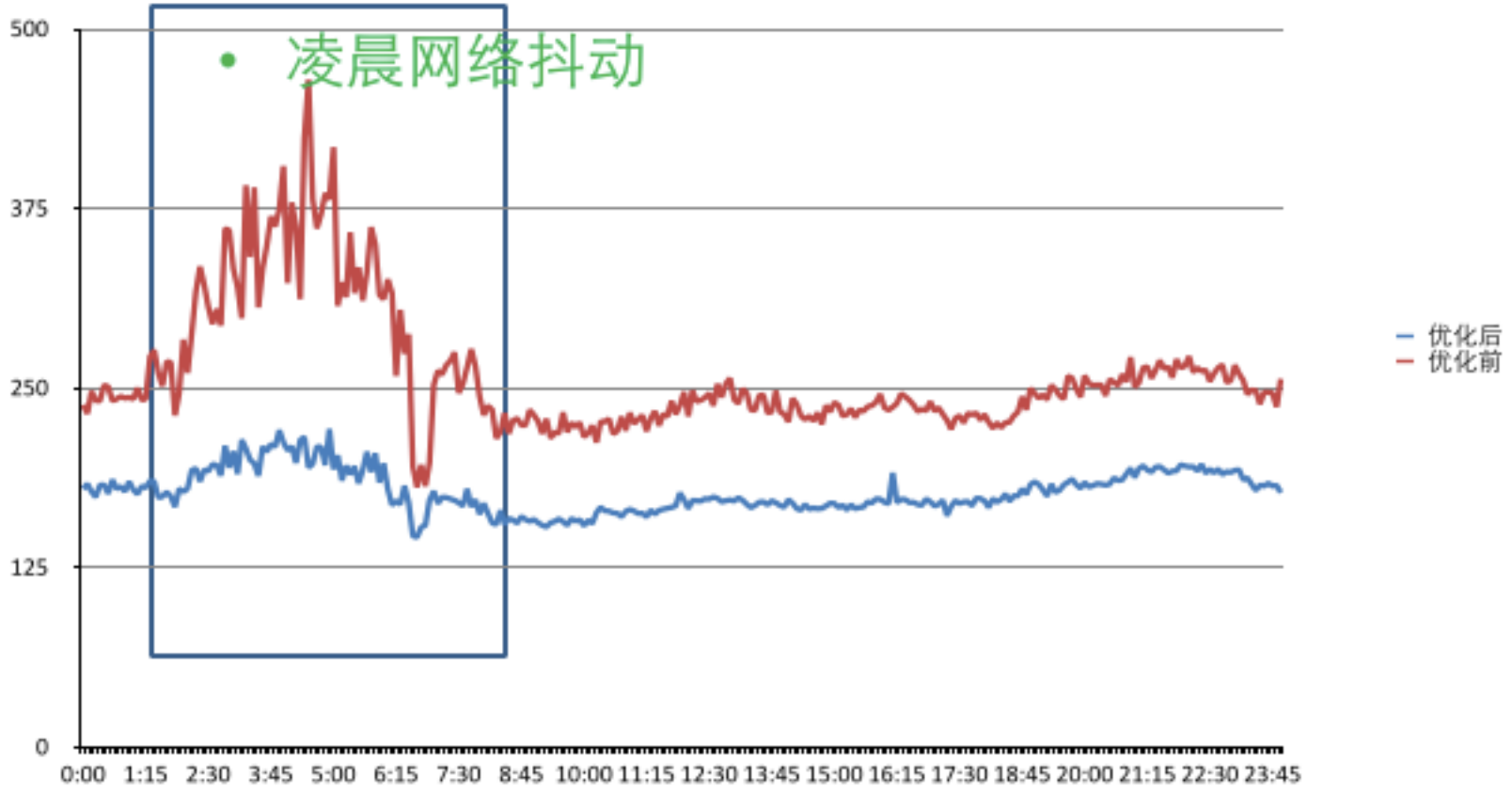
	HTTP	SPDY	对比
QPS	33.5K	33.4K	基本相同
User CPU	15.00	12.83	**14.47%**（优化降低）
Sys CPU	16.20	12.77	**21.17%**（优化降低）
Sirq CPU	10.04	8.48	**15.53%**（优化降低）
Total CPU	41.25	34.10	**17.33%**（优化降低）

TCP协议栈优化

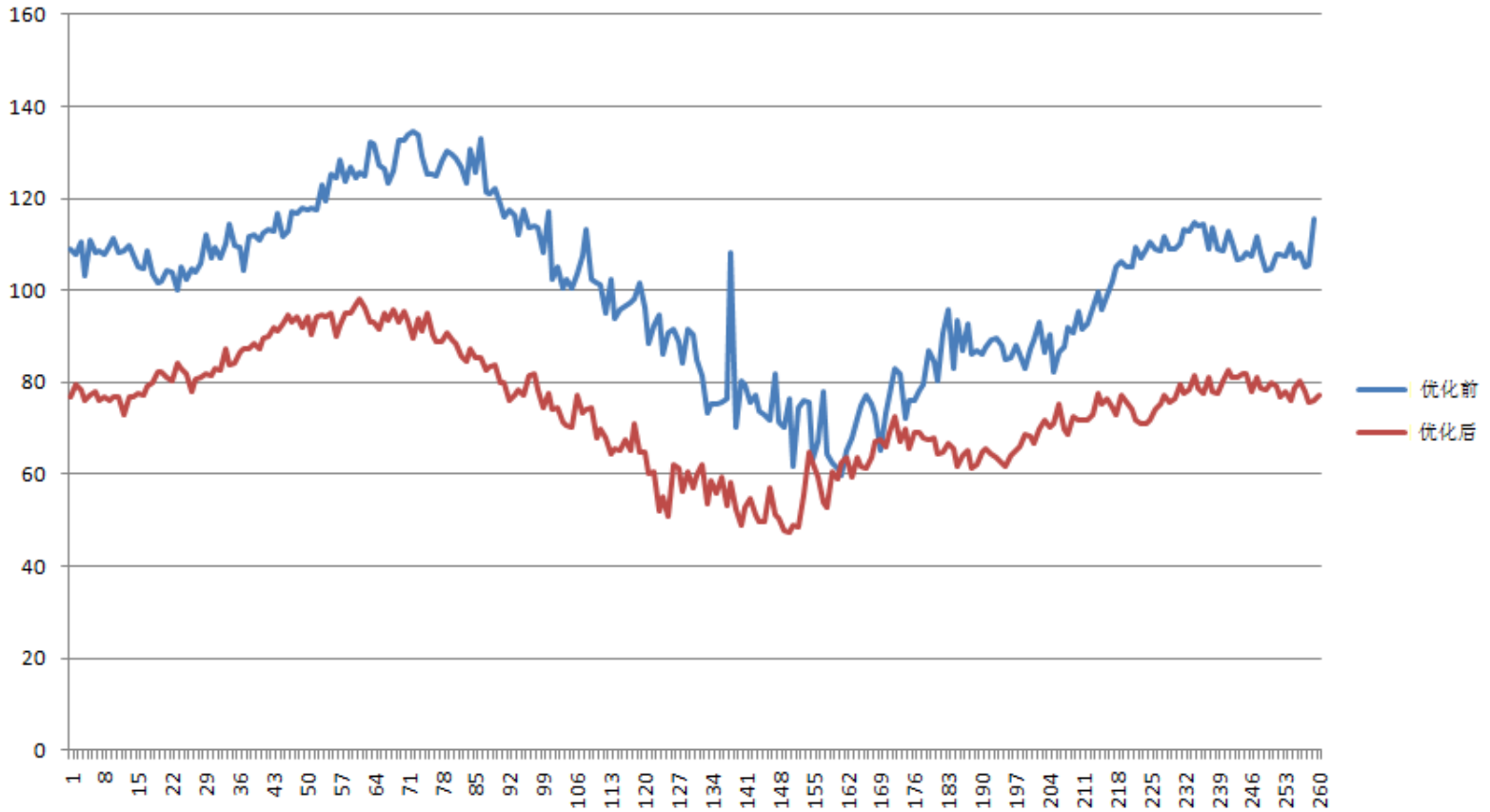
- 改进措施
  - 基于时间序的丢包发现机制
  - 主动的丢包发现机制
  - 自适应的初始窗口
  - 更激进的拥塞避免算法
  - 更小的连接超时时间

	[redacted]	[redacted]	[redacted]	[redacted]
优化前	156ms	600ms	238ms	644ms
优化后	106ms	500ms	174ms	492ms
提升效果	32%	16.5%	27%	24%

TCP协议栈优化效果：抗抖动



TCP优化效果：减少连接时间

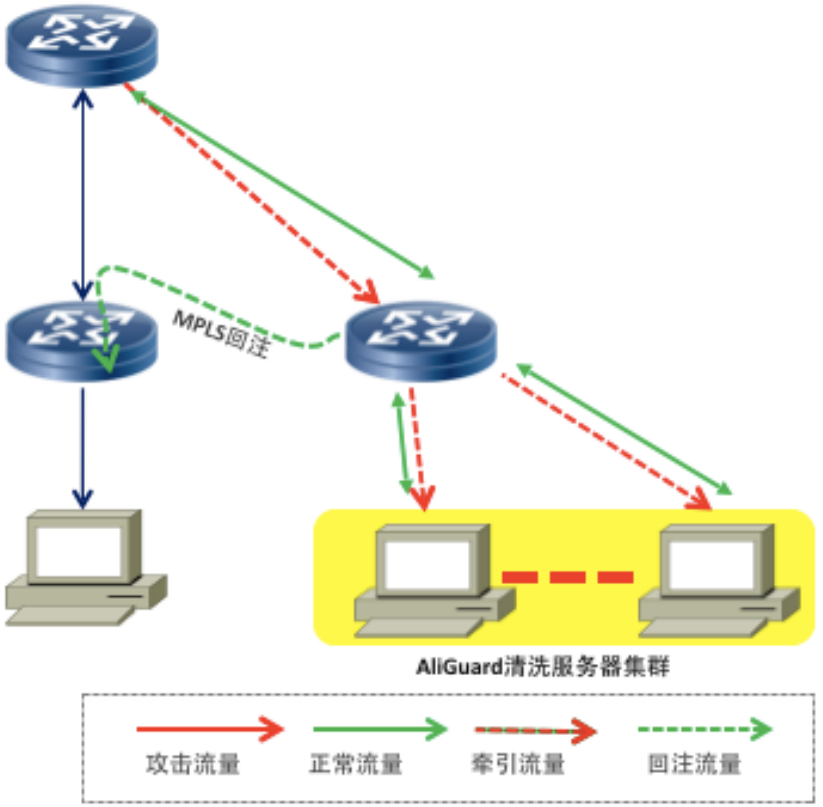


页面优化

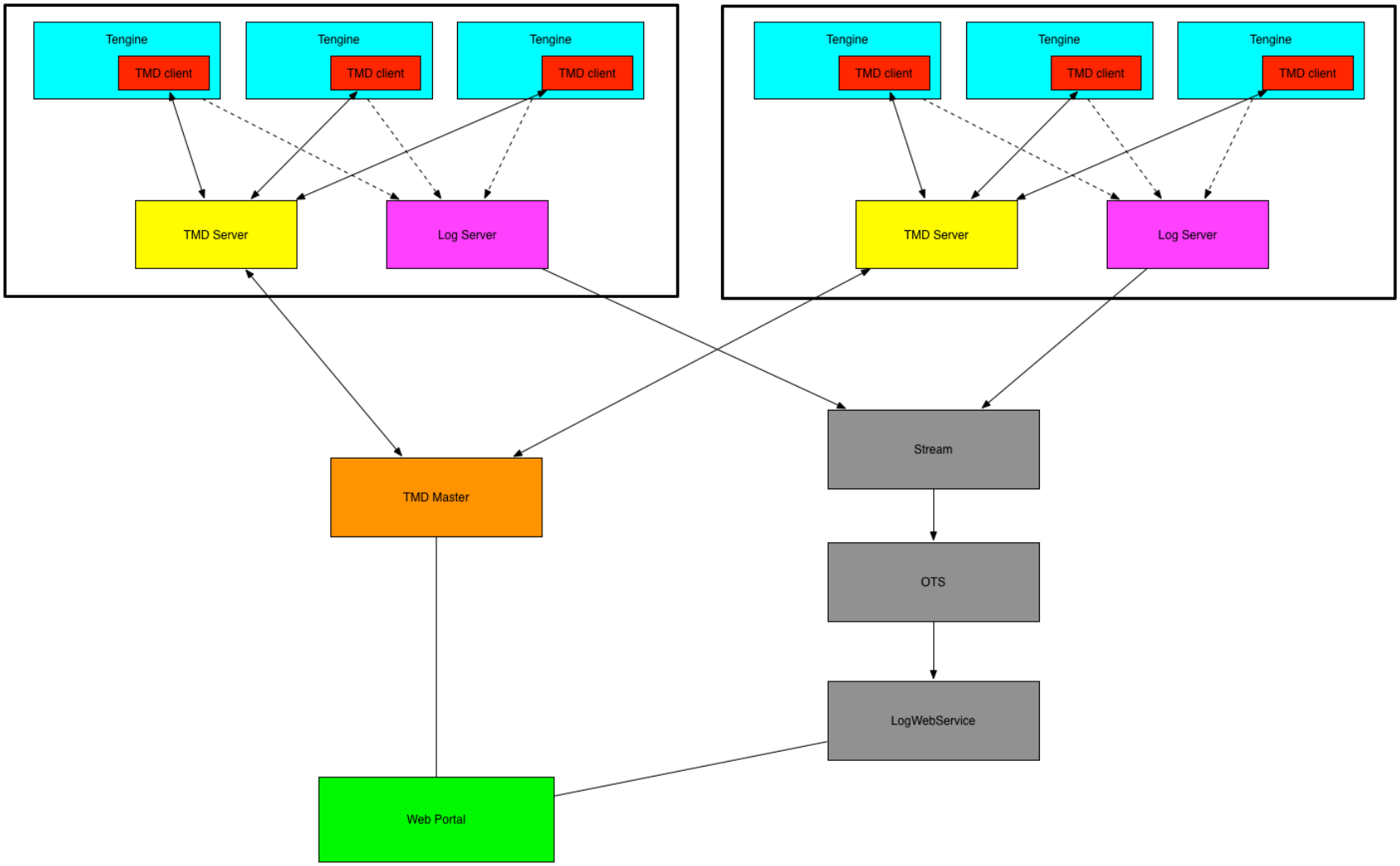
- Trim：去除页面的空格、回车换行、TAB、注释等，以减少页面的大小
- 智能gzip：某些用户的浏览器实际支持gzip但是却被防火墙或者proxy给改掉。智能gzip功能会对这个过程进行测试，从而允许gzip，减少用户传输内容的大小
- SDCH：压缩算法优化，降低传输大小
- Combo：组合多个JavaScript/CSS文件成一个请求，从而减少请求数目

阿里四层防攻击系统：AliGuard

- 基于DPDK之上的网络框架
- 支持集群部署
- 流量牵引
- 四层DDoS攻击防护
- DNS攻击防护



七层防攻击：TMD系统架构



TMD一些关键技术

- 模块化，如防CC模块、hotpatch模块等
- socketpair 实现多进程间配置更新通知
- 共享内存hash表实现黑白名单
- 漏桶，令牌桶算法实现QPS限流
- LRU，红黑树实现CC统计算法
- 多线程，libev实现网络通信框架

TMD防CC攻击的一个例子

- 原页面60KB
- 攻击9万QPS
- 计算带宽41Gbps
- 实际节省200倍

七层防攻击：Web应用防火墙

- 基于Tengine的模块（WAF）
- 高效的规则匹配引擎
- 防止攻击
- SQL注入
- XSS
- Web Shell
- …

可运维性改进

- 海量域名管理
  - Tengine不再依赖配置文件
  - HTTP接口去configserver拿域名对应的配置
  - lazy更新，只记录访问过的
  - 有cache时间
  - 失效接口
  - 不需要reload

我们求贤若渴!

阿里云 aliyun.com

- 一流的技术环境，一流的技术挑战
- 招聘职位
  - 资深CDN系统研发工程师（C/C++）
  - 资深Web服务器开发工程师（C/C++）
  - 资深Java开发工程师
- 欢迎发送简历到
  - 邮件：shudu@taobao.com
  - 新浪微博：@淘叔度
  - 来往：叔度

Thanks!