淘宝网 Taobao.com

SACC 2010系统架构师大会

Taobao
海量图片存储与CDN系统

章文嵩 （正明）
淘宝核心系统部

2010.8.27

2010系统架构师大会

议程

一、系统全貌

二、Taobao图片存储系统--TFS

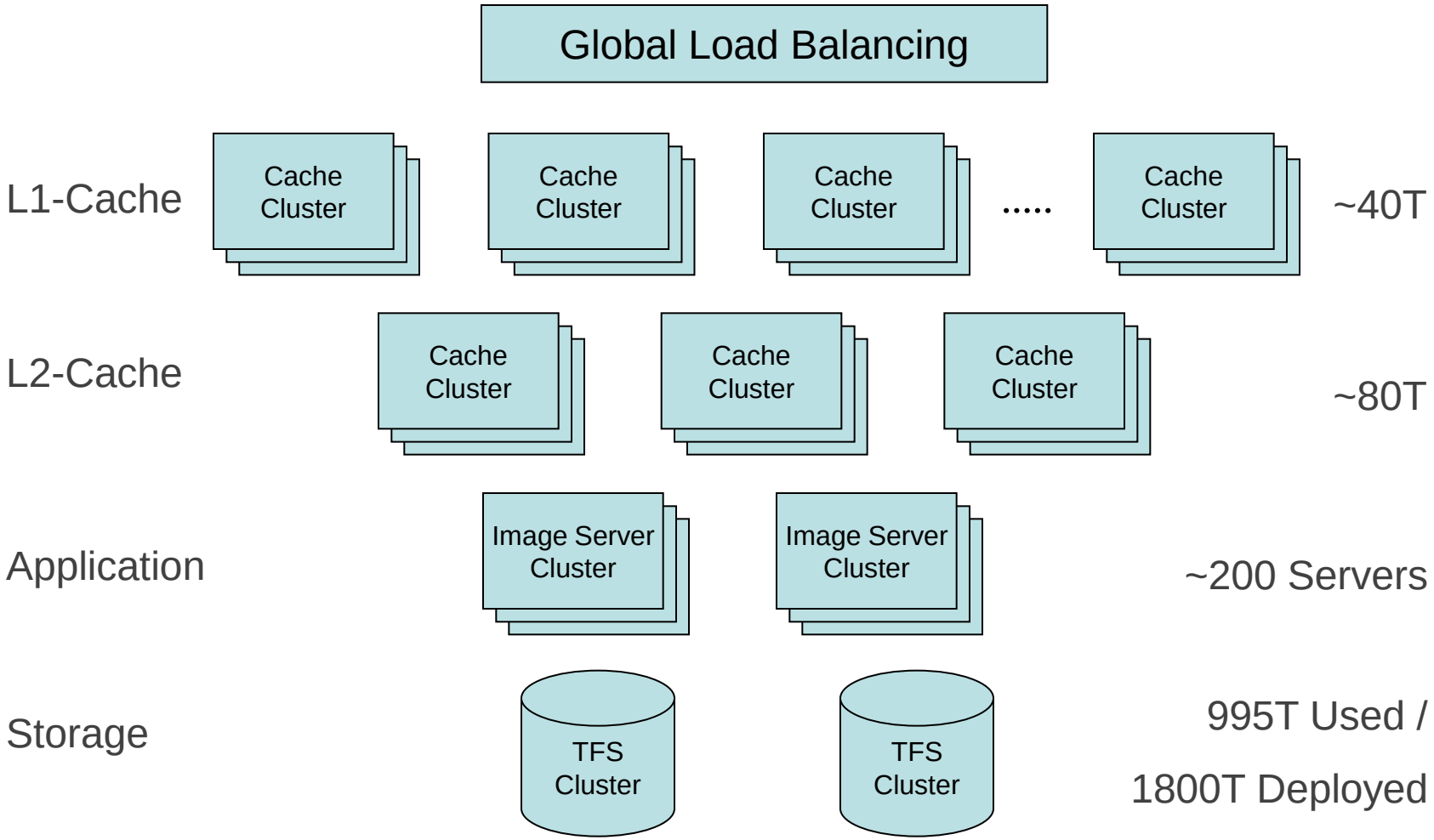
三、Image Server与Cache

四、CDN系统

五、低功耗服务器平台

六、经验

图片存储与CDN系统全貌



议程

一、系统全貌

二、Taobao图片存储系统--TFS

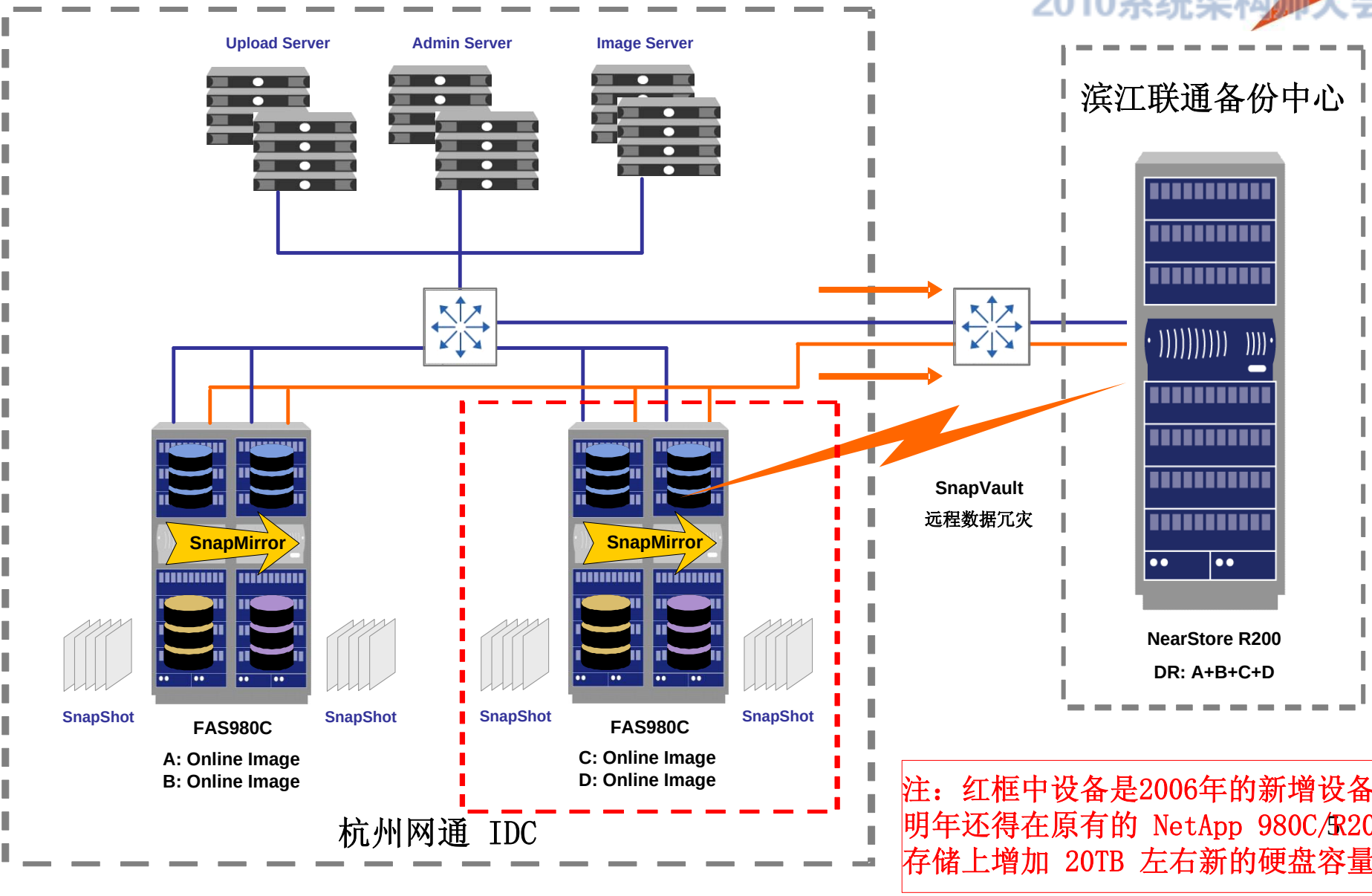
三、Image Server与Cache

四、CDN系统

五、低功耗服务器平台

六、经验

2007年之前的图片存储系统



注：红框中设备是2006年的新增设备，明年还得在原有的 NetApp 980C/R200 存储上增加 20TB 左右新的硬盘容量。

商用产品无法满足需求

- 系统需求
  - 淘宝的影响越来越大，数据的安全也更加重要
  - 数据存储量以每年二倍的速度增长（即原来的三倍）
- 商用存储产品
  - 对小文件的存储无法优化
  - 文件数量大，网络存储设备无法支撑
  - 连接的服务器越来越多，网络连接数已经到达了网络存储设备的极限
  - 扩容成本高，10T的存储容量需要几百万￥
  - 单点，容灾和安全性无法得到很好的保证

TFS 1.0

- 2007年6月

  淘宝自主开发的分布式的文件系统

  TFS（Taobao File System）1.0上线运行

  主要解决海量小文件的分布式存储

  集群规模：200台PC Server(146G*6 SAS 15K Raid5)

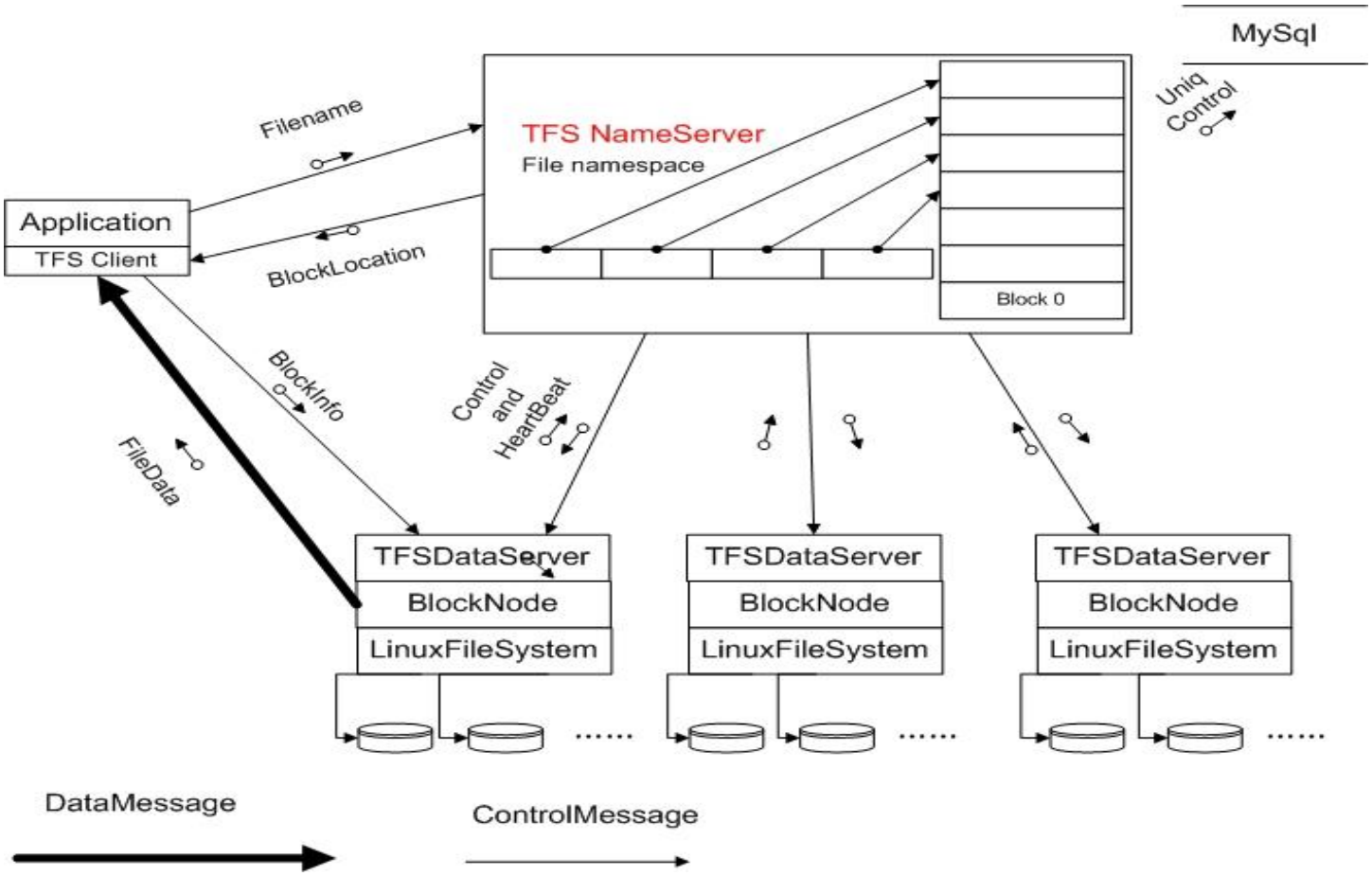
  文件数量: 亿级别

  系统部署存储容量: 140 TB

  实际使用存储容量: 50 TB

  单台支持随机IOPS 200+，流量3MBps

TFS 1.0的逻辑结构



TFS1.0的特性

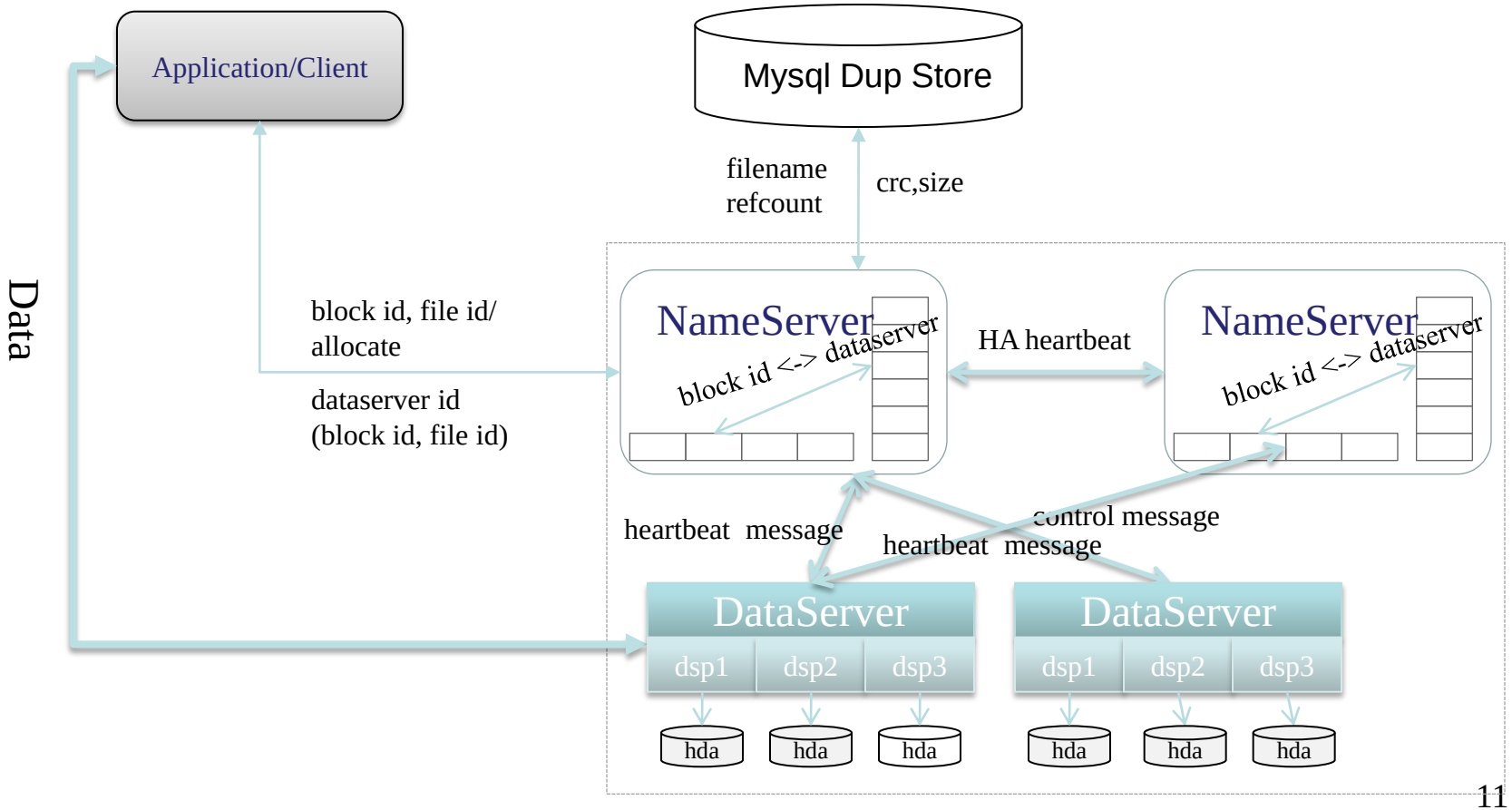
- 集群由一对Name Server和多台Data Server构成
- 每个Data Server运行在一台普通的Linux主机上
- 以block文件的形式存放数据文件(一般64M一个block)
- block存多份保证数据安全
- 利用ext3文件系统存放数据文件
- 磁盘raid5做数据冗余
- **文件名内置元数据信息，用户自己保存TFS文件名与实际文件的对照关系** – 使得元数据量特别小

TFS 1.3

- 2009年6月

  TFS（Taobao File System）1.3上线运行
- 集群规模（2010.8.22）
  - 440台PC Server (300G*12 SAS 15K RPM) + 30台PC Server (600G*12 SAS 15K RPM)
  - 文件数量： 百亿级别
  - 系统部署存储容量： 1800 TB
  - 当前实际存储容量： 995TB
  - 单台Data Server支持随机IOPS 900+，流量15MB+
  - 目前Name Server运行的物理内存是217MB（服务器使用千兆网卡）

TFS1.3的逻辑结构



TFS1.3的特性

- TFS1.3提供了一些重要的功能特性
  - 所有的元数据全部都内存化
  - 清理磁盘空洞
  - 容量和负载的均衡策略
  - 平滑的扩容
  - 数据安全性的冗余保证
  - 几秒内完成Name Server故障自动切换
  - 容灾策略
  - 性能大幅提升

TFS的发展

- TFS2.0正在开发中
  - 大小文件的共存，大文件的分片存储
  - 分级存储机制（SSD/SAS/SATA）的建立，针对访问特性的文件迁移

  ……
- TFS将在9月开源，希望更多人来使用和改进TFS

议程

一、系统全貌

二、Taobao图片存储系统--TFS

三、Image Server与Cache

四、CDN系统

五、低功耗服务器平台

六、经验

图片处理与缓存

- 现状
  - 有200多台服务器Image Server，在Apache上实现的，从TFS取原图生产相应的缩略图
- 改进目的
  - 图片访问的热点一定存在，在Image Server实现Cache，提高响应速度，也减轻对后端TFS的压力
- Image Server上处理方式
  - 若请求图片在Cache中，直接发送
  - 没命中，若本地有原图，则根据原图做处理并缓存
  - 没命中，从TFS读取原图并添加到缓存，处理并缓存

系统实现

- 将图片处理与缓存编写成基于Nginx的模块
  - Nginx是目前性能最高的HTTP服务器（用户空间）
  - 代码清晰
  - 模块化非常好
- 使用GraphicsMagick进行图片处理
  - 比ImageMagick性能更好
- 面向小对象的缓存文件系统
- 前端有LVS+Haproxy将原图和其所有缩略图请求都调度到同一台Image Server

图片处理

- 从TFS存储中读取文件
- 将文件根据需要的尺寸进行缩放
  - 灵活，应用可以制定一些尺寸规则决定
  - 动态计算的成本大概是存储缩略图的十分之一
- 可根据需要将缩略图按一定质量压缩保存（75%～94%）和锐化处理，通过配置文件设定
  - 权衡图片的效果与CDN传输的带宽
  - 低损压缩降低缩略图的体积（30～70%）

本地缓存对象文件系统

- 文件定位
  - 内存hash做索引
  - 最多一次读盘
- 写盘方式
  - Append方式写
  - 淘汰策略FIFO，主要考虑降低硬盘的写操作，没有必要进一步提高Cache命中率，因为Image Server和TFS在同一个数据中心

议程

一、系统全貌

二、Taobao图片存储系统--TFS

三、Image Server与Cache

四、CDN系统

五、低功耗服务器平台

六、经验

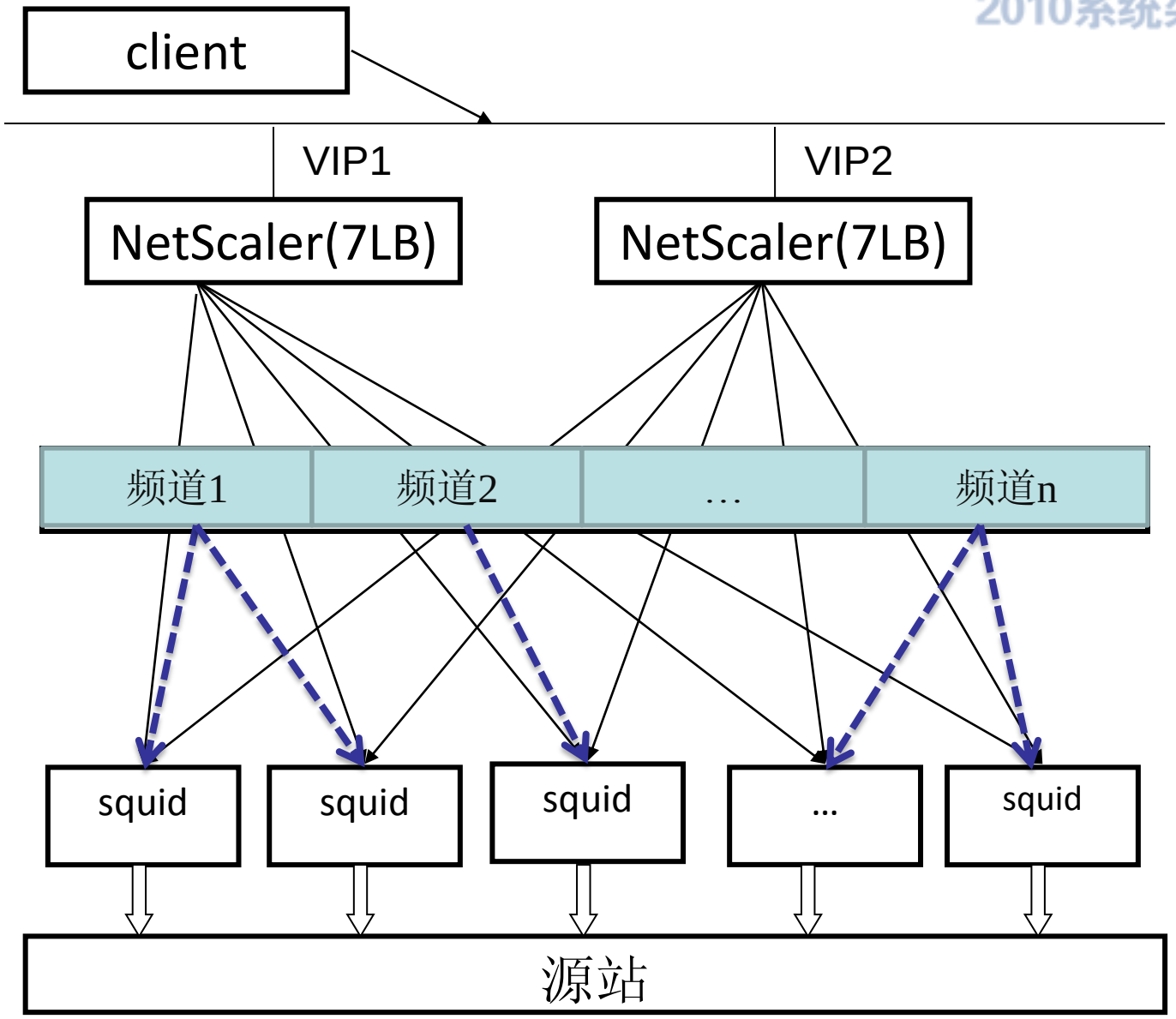
淘宝CDN系统

- CDN服务的图片规模
  - 约250T容量的原图 + 250T容量的缩略图
  - 约286亿左右的图片数，平均图片大小是17.45K
  - 8K以下图片占图片数总量的61%，占存储容量的11%
- CDN部署规模
  - 22个节点，部署在网民相当密集的中心城市(7月初)
  - 每个节点目前处理能力在10G或以上
  - CDN部署的总处理能力已到220G以上
  - 目前承载淘宝流量高峰时119G，含一些集团子公司的流量

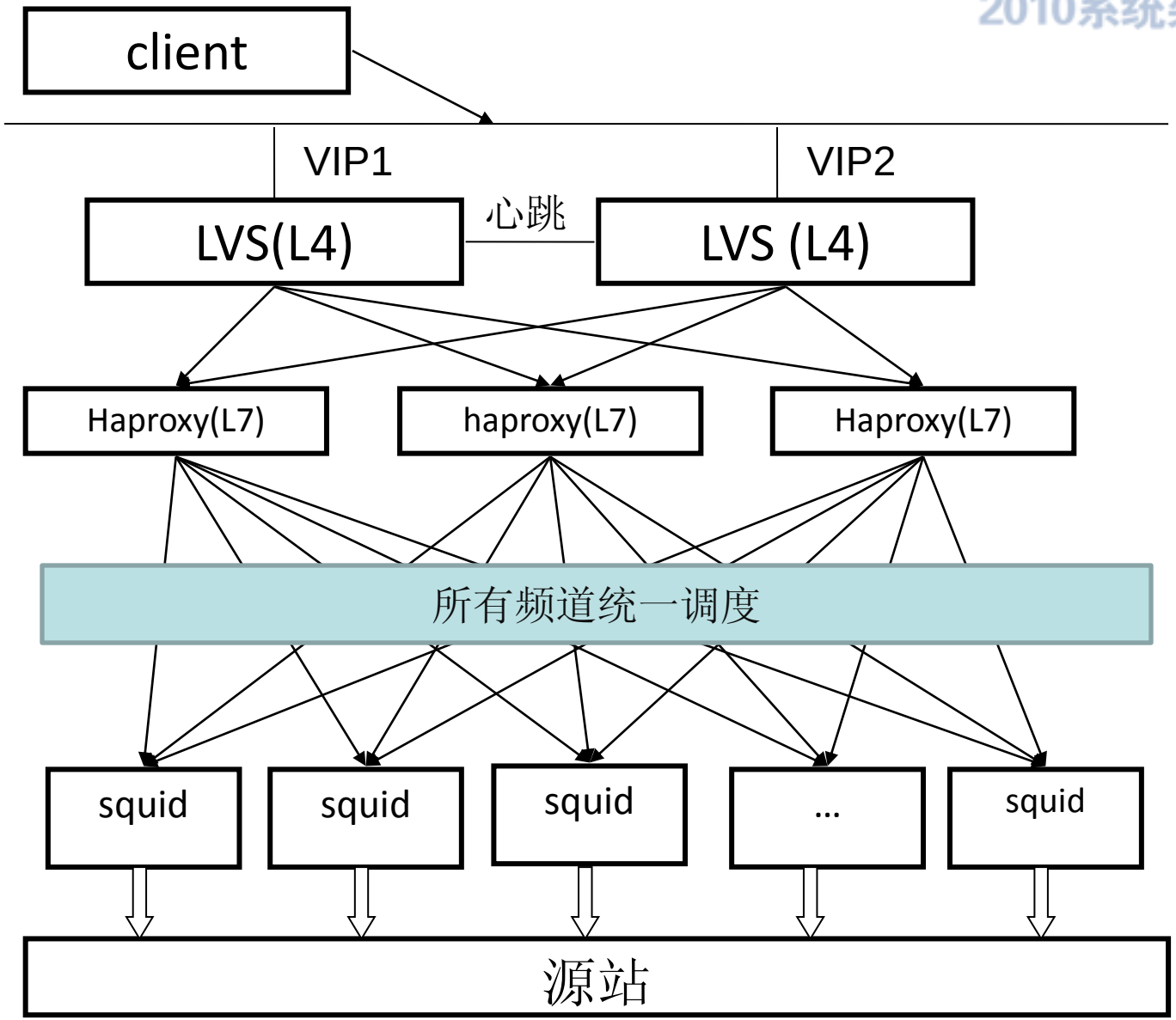
淘宝CDN系统的研发

- 主要解决现有的问题
  - 商用产品的性能瓶颈、功能欠缺，以及不稳定性
  - 整个系统的规模、性能、可用性和可管理性
- 开发完全自主的CDN系统
  - CDN节点的新架构和优化
  - CDN监控平台
  - 全局流量调度系统支持基于节点负载状态调度和基于链路状态调度
  - CDN实时图片删除
  - CDN访问日志过滤系统
  - 配置管理平台

CDN节点的架构对比-老架构



CDN节点的架构对比-新架构



CDN节点的架构对比

对比项 \ 节点	新架构	老架构
流量分布均匀性	☆☆☆☆☆	☆☆☆
可维护性	☆☆☆	☆☆☆
抗攻击能力	☆☆☆☆	☆☆☆☆
自主控制能力	☆☆☆☆☆	☆☆☆
价格	☆☆☆☆☆	☆☆☆
扩展能力	☆☆☆☆☆	☆☆
灵活性	☆☆☆☆☆	☆☆

- 流量分布均匀性：所有的频道统一调度到128台squid，而不是将squid按频道分组，可提高命中率2%以上
- 扩展能力：在一个VIP上新架构可以扩展到近100G的流量（当然要用万兆网卡）
- 灵活性：一致性Hash调度方法使得增加和删除服务器非常方便，只有1/(n+1)的对象需要迁移

Squid改进和优化

- 在COSS存储系统基础上实现了TCOSS，FIFO加上按一定比例保留热点对象，支持1T大小的文件
- Squid内存优化，一台Squid服务器若有一千万对象，大约节省1250M内存，更多的内存可以用作memory cache
- 用sendfile来发送缓存在硬盘上的对象，加上page cache，充分利用操作系统的特性
- 针对SSD硬盘，可以采用DIRECT_IO方式访问，将内存省给SAS/SATA硬盘做page cache
- 在Squid服务器上使用SSD+SAS+SATA混合存储，实现了类似GDSF算法，图片随着热点变化而迁移

$$migration\_weight * \frac{frequency}{size^{migration\_power}} \; ; \quad migration\_power \in (0,1]$$

热点迁移的数据对比（1）

- 简单按对象大小划分：小的进SSD，中的放SAS，大的存SATA
- SSD + 4 * SAS + SATA上的访问负载如下：

```
[root@cache161 ~]# iostat -x -k 60 | egrep -v -e "sd.[1-9]"
…
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          3.15    0.00    5.63   11.35    0.00   79.87

Device:         rrqm/s   wrqm/s   r/s   w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
sda            15.40     1.17 50.66  2.63  2673.22   124.85   105.01     0.55   10.39   6.27  33.41
sdb             0.07     0.03 447.29  1.02  4359.01   191.90    20.30     0.32    0.71   0.27  12.13
sdc             5.73     1.53 114.93  8.42  1264.86   100.58    22.14     1.05    8.48   3.56  43.94
sdd             5.57     2.07 121.83  9.57  1319.45   104.12    21.67     1.19    9.02   3.63  47.72
sde             5.53     1.45 111.45  8.52  1246.53   101.92    22.48     0.95    7.88   3.42  41.06
sdf             5.45     2.02 118.93  8.00  1281.92   106.25    21.87     1.19    9.37   3.74  47.44
```

其中：黑色为SATA，绿色为SSD，红色为SAS
4块SAS硬盘上的访问量和超过SSD硬盘上的访问量

热点迁移的数据对比（2）

- 按对象访问热点进行迁移：最热的进SSD，中等热度的放SAS，轻热度的存SATA
- SSD + 4 * SAS + SATA上的访问负载如下：

```
[root@cache161 ~]# iostat -x -k 60 | egrep -v -e "sd.[1-9]"
…
avg-cpu:  %user   %nice %system %iowait  %steal   %idle
          3.15    0.00    5.63   11.35    0.00   79.87

Device:         rrqm/s   wrqm/s   r/s   w/s    rkB/s    wkB/s avgrq-sz avgqu-sz   await  svctm  %util
sda               5.08     1.65 18.55  2.52  1210.07   119.00   126.18     0.14    6.50   5.46  11.51
sdb               1.68     0.05 610.53  1.75  6962.29   413.47    24.09     0.28    0.46   0.23  14.25
sdc               0.22     0.03 28.87  0.97  1172.93   189.13    91.31     0.16    5.28   4.40  13.13
sdd               0.23     0.02 29.70  0.77  1133.47   122.53    82.45     0.15    4.99   4.39  13.37
sde               0.18     0.03 28.23  1.03  1078.73   206.27    87.81     0.15    5.00   4.24  12.40
sdf               0.10     0.02 28.42  0.55  1090.27   115.00    83.22     0.15    5.04   4.44  12.86
```

其中：黑色为SATA，绿色为SSD，红色为SAS
SSD硬盘上的访问量是4块SAS硬盘上访问量之和的5倍以上，SAS和SATA的硬盘利用率低了很多

节点运行情况（1）

- 节点规模：32台 DELL R710服务器
- 逻辑结构：2 LVS + 32 Haproxy + 64 Squid
- 时间：12月21日上线运行
- 当前最大服务流量：10.58 Gbps
- 理论最大负载能力：15Gbps以上
- 单台R710服务器可到500Mbps以上的吞吐率
- 单squid最大object数目：1800万
- Cache请求命中率：97%
- Cache字节命中率：97%
- **最重要的是命中率提高，大大改善用户的访问体验**

节点运行情况（2）

- 节点规模：30台 DELL PowerEdge 2950服务器
- 逻辑结构：2 LVS + 30 Haproxy + 60 Squid
- 时间：2010年5月上线运行
- 理论最大负载能力：12Gbps
- 单台2950服务器可到400Mbps的吞吐率
- 单台存储：160G SSD + 143G SAS * 4 + 1T SATA
- 单squid最大object数目：3000万
- Cache请求命中率：97.5%
- Cache字节命中率：97.5%
- **最重要的是命中率提高，大大改善用户的访问体验**

CDN系统的发展

- CDN系统的研发与运维
  - 针对教育网的CDN解决方案
  - 动态页面加速，节点间应用级路由
  - 持续提高节点性能（应用软件、操作系统等）
  - 优化GTM全局调度系统
  - 持续提高CDN系统可运维性，完善CDN内容管理系统
- CDN系统的建设
  - 思路正在转向“部署更多的小节点，尽可能离用户近一些”
  - 定制化和快速部署

议程

一、系统全貌

二、Taobao图片存储系统--TFS

三、Image Server与Cache

四、CDN系统

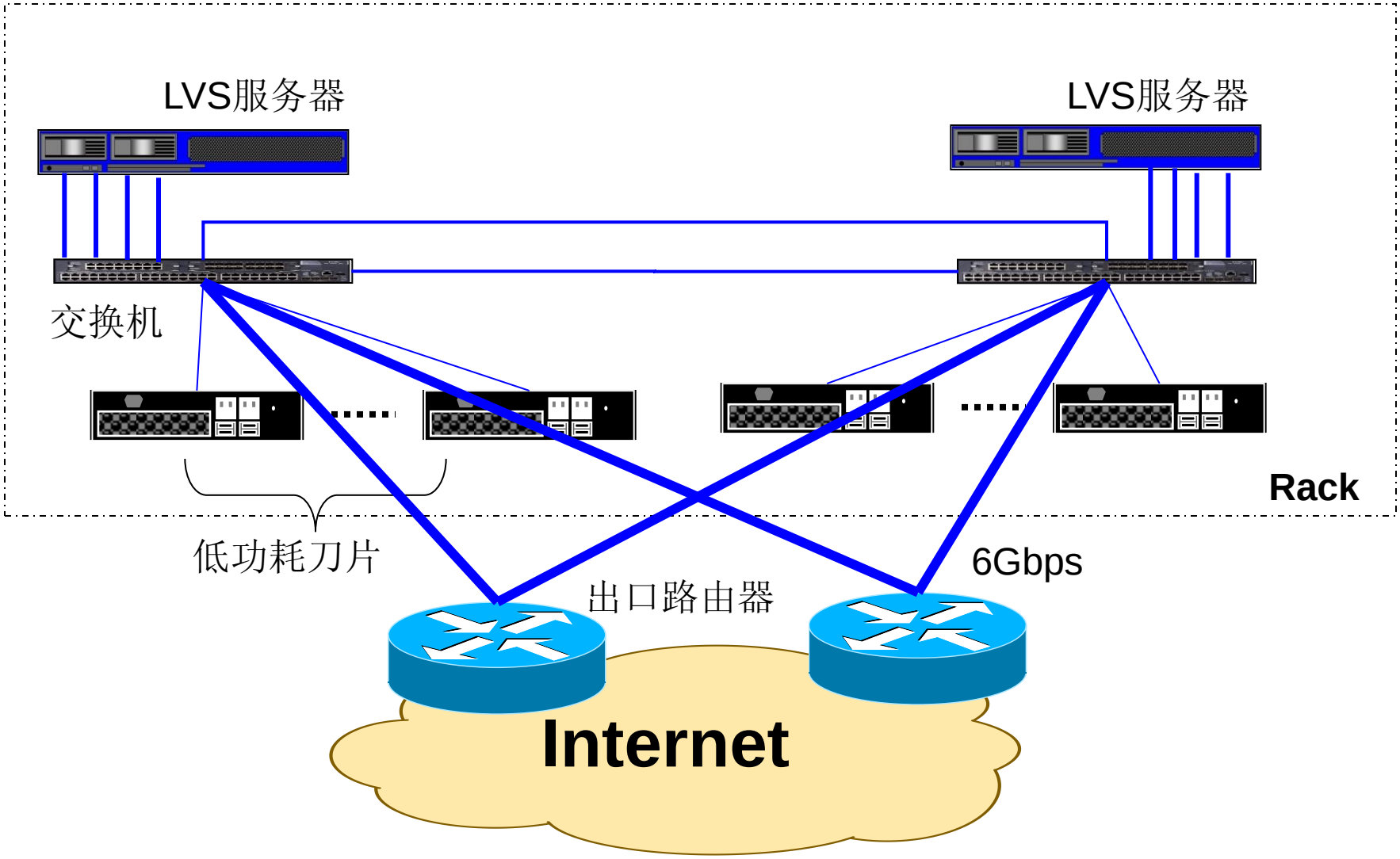
五、低功耗服务器平台

六、经验

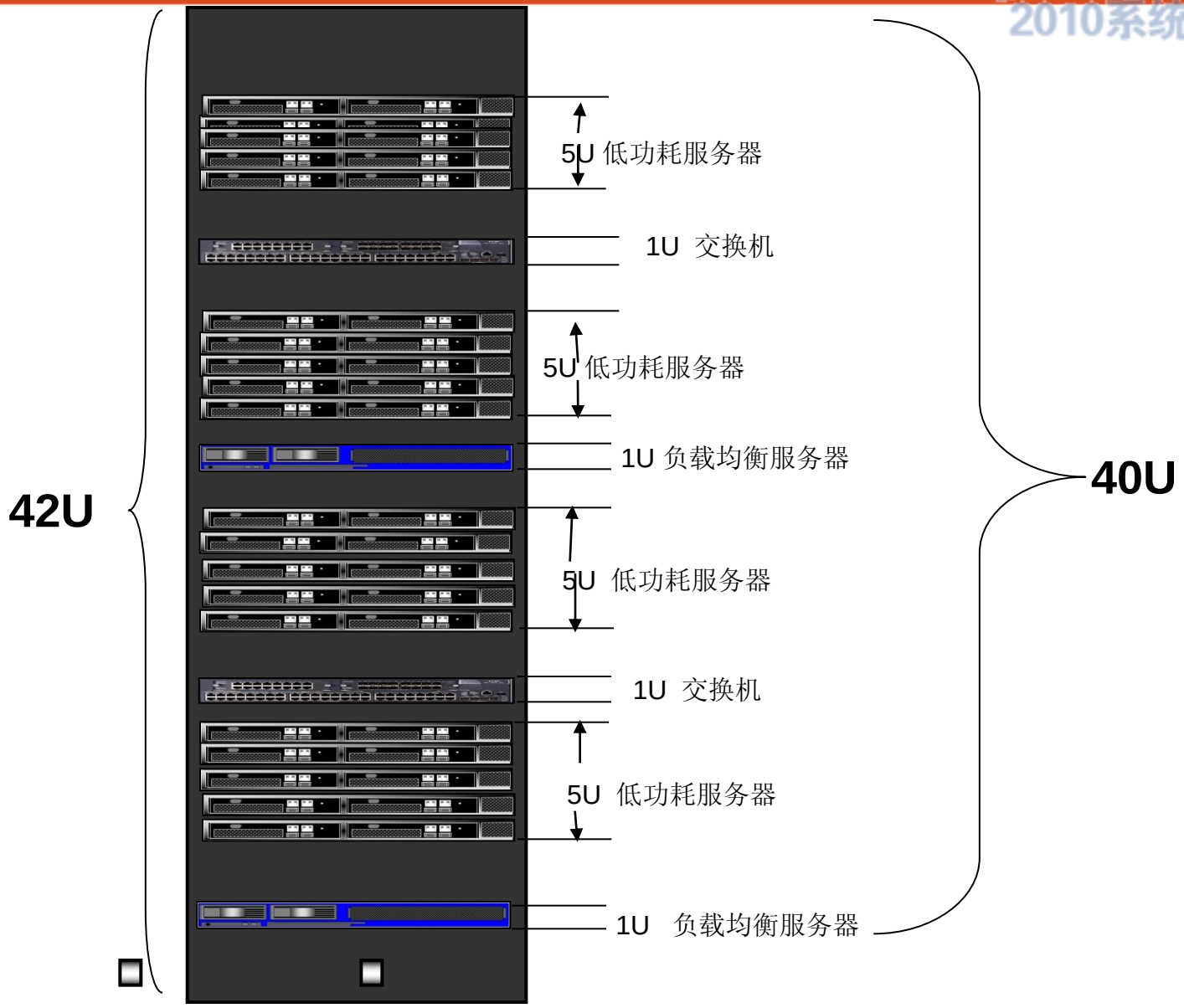
低功耗硬件平台

- 低功耗硬件平台
  - 低功耗的CPU，如Intel ATOM, VIA Nano等
  - 低功耗的Chipset；SSD或低功耗的SATA硬盘
  - 关闭GPU和USB Controller等
- 适用不需要太多CPU计算的I/O类型应用
  - 例如CDN Cache Server、memory cache、存储节点、静态文件Web Server等
- 好处（大大降低成本）：
  - 降低电力消耗，减少碳排放
  - 单位空间(机柜)下有更高的I/O吞吐率
  - 降低硬件购置成本和运营成本

单机柜方案:节点网络结构



单机柜(6Gbps)方案:节点机架布局



议程

一、系统全貌

二、Taobao图片存储系统--TFS

三、Image Server与Cache

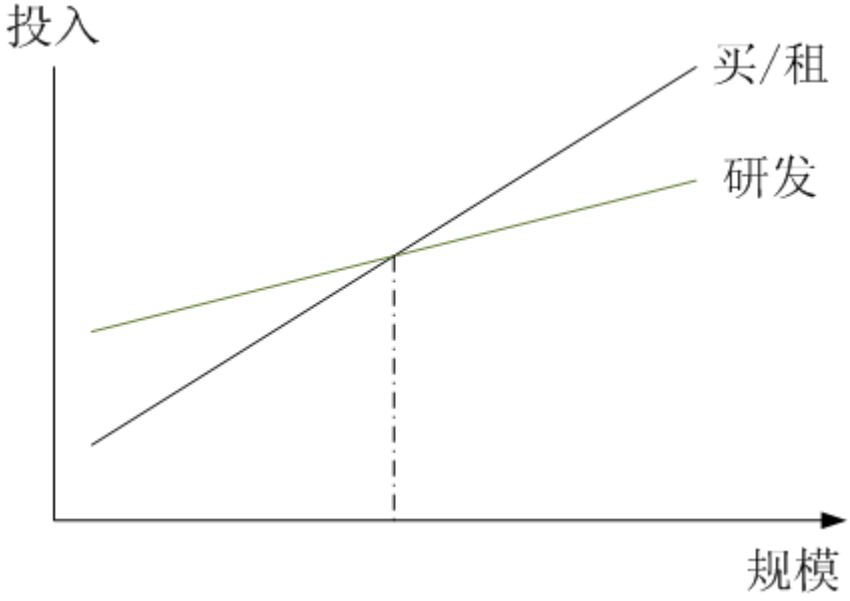
四、CDN系统

五、低功耗服务器平台

六、经验

经验

- 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，系统上有更高的可扩展性
- 规模效应，研发投入都是值得的
- 可以在软件和硬件多个层次优化
- 优化是长期持续的过程



淘宝开源平台

淘宝开源

- Code.taobao.org是开放的开源平台，淘宝公司在上面发布开源项目，也非常欢迎外部人员在上面发布开源项目
- 平台本身的管理软件也作为一个开源项目
- 淘宝的Key/Value Cache/Store - TAIR已经开源
- 会陆续将淘宝的基础软件开源
  - TFS将于9月开源
  - WebX v3.0将于10月开源
- 淘宝公司希望以更开放的方式与业界一起进行技术创新

Q & A

谢谢!