

马哥Linux运维实战项目进阶



该案例为内部机密，请勿外传

马哥Linux运维实战学院出品

项目描述

Project description

项目需求

近年来为适应业务发展的需求，世界500强XX企业准备进行大规模的电子商务建设，同时，希望能通过Linux平台，利用开源技术，来实现大型互联网电子商务网站架构建设和业务支撑，现要求成立运维技术保障部门，并邀请你担当运维部门经理，对整个运维部门进行部署和规划。当你拿到该项目时，需要做以下几方面事情：



整体项目分为四个大阶段：

- 一、前期筹划阶段
- 二、项目实施阶段
- 三、项目升级改造阶段
- 四、项目验收和维护阶段

一、前期筹划阶段

1、业务需求研究	充分了解业务需求，从架构、可用性、并发、峰值、可维护性了解业务
2、人员分工规划	根据业务场景，和人员技能上，分配3类角色，分别是运维架构师，系统工程师，应用运维工程师。进行项目团队合作。
3、服务器选型	根据业务需求，对服务器硬件进行选型，对CPU、内存、硬盘、等数据进行统筹。
4、基础系统、软件选型	对操作系统类型，版本，和常用开源软件，以及版本进行选型
5、架构方案设计	根据业务场景，设计出对应的运维架构框架，从用户引导层、负载均衡层、缓存层、应用层、数据层、管理端、监控端等多维度出发
6、初步可实施方案探究	对设计的方案进行可行性研究，验证架构的可行性，从业务的质量、维护成本、可用性、可扩展性等多角度进行衡量

1、业务需求描述：

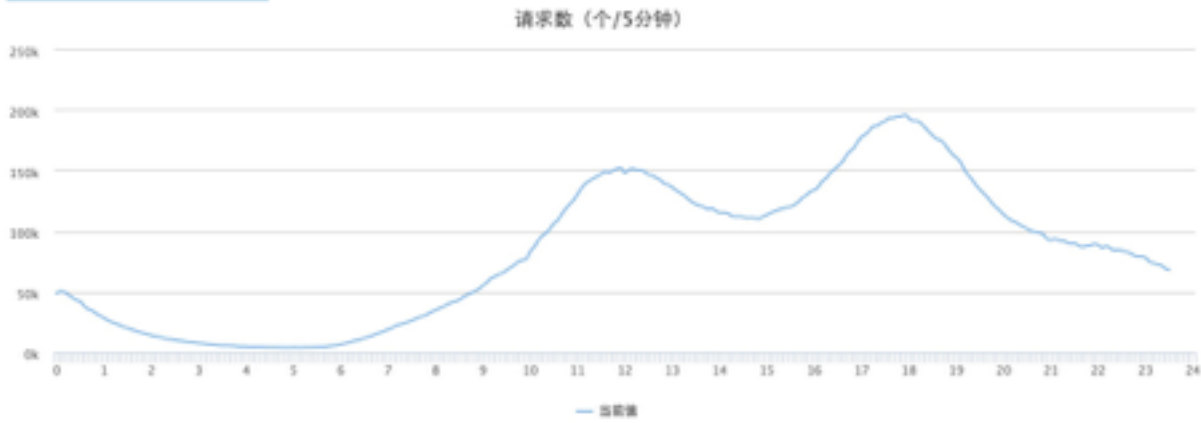
本公司计划组建电子商务子公司，计划三个月内进行业务上线，上线前，需要完成人员配置、服务器选型、基础系统、软件选型、架构方案设计初步可实施方案等相关准备工作。

本公司上线初期，每日**PV300W**，总请求**3000W**，**QPS**并发日均值**300**，峰值**900**，先实现本业务量架构和基础服务。

上线**1**个月后，集团会大力推广相关业务，预计三个月后将达到每日总请求**PV2500W** 总请求数**2.5**亿 **QPS**日均为**2000**，**QPS**峰值**6000**。

指标对比

请求数 成功率 成功耗时



类别	成功率(%)	总请求数	成功平均延迟(ms)	平均发包(包)	平均回包(包)
当前值	99.783%	23,603,916	683.329	0	2,022

点击展开，进行OLAP查询

网络类型	版本	连接类型	平台	地区	运营商	来源	成功率(%)	总请求数	成功平均延迟(ms)	平均发包(包)	平均回包(包)
展开	展开	展开	展开	展开	展开	展开	99.783%	23,603,916	683.329	0	2,022

假设单台server能承载200QPS，单台缓存能承载2000QPS，缓存命中率90%，单台数据库能承载5000QPS，每次QPS都包含一次数据库查询。

目前需要给出两套方案，前期方案和后期方案：

前期方案：适应于初期上线阶段，架构容量能保证峰值的2倍，能支撑公司正常业务，有基础设施，保证正常上线下流程。

后期方案：在前期方案基础上，进行改造，适用于推广后阶段，需要保证架构容量能保证峰值的2倍。全部应用无单点，要求有测试环境，全部系统无单点，并要求架构可快速扩容，有监控、管理、备份方案和策略，需要降级方案，切换方案，上线流程方案。

二、项目准备工作

1、人员分工规划

架构师	负责整体规划，项目把控，技术选型和业务落地
系统工程师	负责硬件选型，系统初始化，系统优化，目录标准化，基础应用搭建，监控等
应用运维工程师	负责管理业务，完善应用配置文件，应用层监控，日志收集，故障排错

2、硬件和软件选型

硬件选型	对于不同应用选择不同硬件资源进行匹配，如mysql需要存储空间，缓存需要内存等
系统选型	选择合适的操作系统
软件选型	对使用的开源软件进行选型，如apache nginx php mysql lvs haproxy等
备份选型	包含备份方案，备份策略，备份服务器安全等

3、架构设计

负载均衡层	Haproxy & LVS & Nginx
缓存层	Varnish & Nginx & Squid
应用层	Nginx & Apache & Lighttpd
数据库层	Mysql & Oracle & PostgreSQL
管理工具	Ansible & Puppet & Salt
监控工具	Zabbix & Nagios & Cacti

第一阶段业务支撑架构图



二、项目实施阶段

项目实施步骤（先画出自己的架构图）：

- 1、服务器信息录入（IP、内存、CPU、硬盘）
- 2、操作系统选型和批量安装（系统安装）
- 3、系统初始化（关闭不需要的服务、做安全控制、建立对应目录、调整系统核心参数）
- 4、根据服务器角色，安装应用（根据选择角色进行服务）

- 5、对应用进行和配置和管理
- 6、代码上线和服务器启动
- 7、对应服务确认，确保业务正常（服务可访问，业务正常）
- 8、所需监控项目添加（系统和应用）
- 9、前端业务规则配置（负载均衡配置后端逻辑）
- 10、对外提供服务（对外可提供服务）

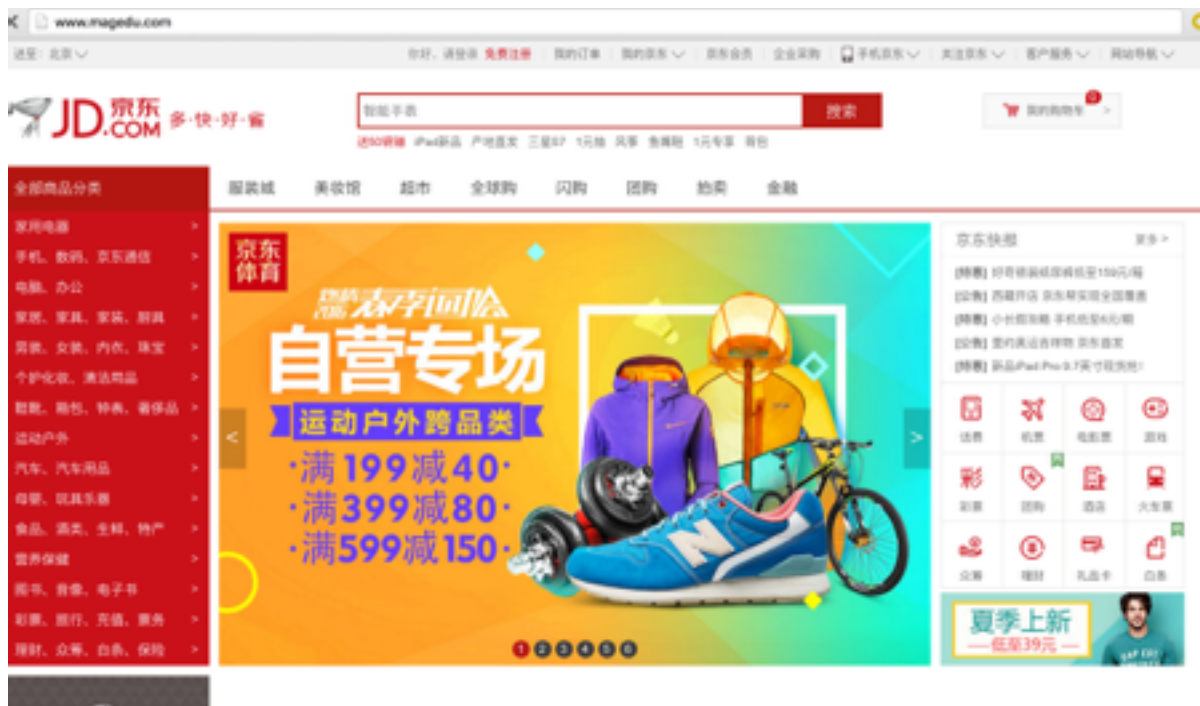
要求：

- 1、全部机房服务器跟时钟服务器进行时钟同步
- 2、全部机房服务器都有自己的主机名，主机名为角色-应用名01、02....N.magedu
如（lb-haproxy01.magedu、web-nginx03.magedu）
- 3、所有主机名可通过机房DNS服务做解析，跳板机可无密码登陆机房其他所有机器
- 4、软件放到/usr/local/软件名目录中，/etc/下有软连接配置文件，数据都放到/data/目录下，如网页数据放到/data/webapps/，日志放到/data/logs/，如/data/logs/nginx.log，所有应用尽可能源码安装，没有的可官网download

- 5、图片需要放到单独服务器，前端负载均衡做规则分离
- 6、监控需要监控网卡流量、内存、**CPU**、磁盘使用、**load**、应用软件状态等指标
- 7、**Mysql**每周日夜里3点做一次全量备份，周一到周六每天夜里2点做一次增量备份。本地备份完，放到备份服务器上。确保放成功后，将本地备份删除
- 8、**Haproxy**对于业务实现应用层监控，基于**http**头部信息监控
- 9、可通过**www.magedu-xxx**(小组名缩写).com访问到对应网站和内容
- 10、要求用**Puppet**管理web server的**nginx.conf**、**php.conf**配置文件
- 11、用**Ansible**或者**shell**脚本批量对线上**Web**服务可进行代码更新，更新策略为一台一台更新
- 12、所有服务，都需要开机自动启动，并有服务管理脚本

13、/data/logs目录下所有日志每天夜里1点，以**.log**结尾的进行压缩归档，并以压缩全格式时间命名。

当完成以上阶段，可看到仿京东商城网站（如下图）。



至此，初期阶段搭建完毕，可通过上线脚本，批量对程序进行更新，更新内容为**./includes/lib_main.php**，将标题换成马哥教育**-magedu.com**，更新过程要求为全量、修改软链方式。（注意上线流程）

可用压测工具如：**Jmeter**（推荐）、**ab**、**Webbench**等常规工具，去测试首页、列表页、详情页的单台容量，系统压测数据可从监控**Zabbix**上获取，评估出来以下数据：

压测评估	并发数和总请求数	平均响应时间\95线	负载和系统资源消耗
首页			
列表页			
详情页			
综合测试			

三、项目升级改造阶段

公司电商网站原只有**每日PV300W，总请求3000W**的业务量，随着集团业务要高速发展，预计三个月后，网站数据情况将增长为**PV2500W 总请求数2.5亿的中型互联网规模网站，所需服务器200+**

此时，网站程序需要进行一次大改版，老板也要求你对运维架构进行一次全面改造和业务升级，来满足新的业务场景。

要求：在前期方案基础上，进行改造，适用于推广后阶段，需要保证架构容量能保证峰值的2倍。全部应用系统无单点，要求有测试环境、压测环境，核心节点主从切换方案，并要求架构可快速扩容，有监控、管理、备份方案和策略，需要有降级方案，回滚方案，上线流程等方案。

扩容项目清单	
测试环境	有单独的测试数据库和线上代码测试环境
压测环境	有专业的压测环境，对网站核心指标进行压测
容量评估	根据压测报告，可评估单台服务器压力以及整体架构容量
全系统无单点	需要对所有系统、应用做高可用方案，保证业务无单点风险
架构可快速扩容	所有架构可快速扩容，应用包和业务代码可以快速上线
监控系统	需要从业务、应用、系统、网络四个层面对体系做全方位监控
管理系统	要求有运维管理系统，对线上业务进行自动化运维管理

扩容项目清单

跳板机	可部署马帮前学员老广的Jumpserver跳板机项目
备份系统	对网站代码，数据库数据提供专门的备份策略
降级方案	当网站不可用时，给出降级策略（保证首页可访问）
回滚方案	当代码上线出现故障，如何进行快速回滚
上线流程	实现专业上线方案，保证项目无缝上线

扩展项目架构图:



项目要求:

按上述描述, 需要完成**PV2500W 总请求数2.5亿, QPS峰值6000**, 的中型互联网规模网站架构, 假设单台server能承载200QPS, 单台缓存能承载2000QPS, 缓存命中率90%, 单台数据库能承载5000QPS, 每次QPS都包含一次数据库查询。

- 1、网站需要实现首页、列表页、详情页的缓存策略。通过**varnish**来抗高并发请求, 缓存策略可用内存方式。
- 2、图片做分离需要缓存策略, 并通过**varnish**进行缓存, 缓存策略可用硬盘映射方式。
- 3、图片需要用分布式**mogileFS**做存储, 做2份副本。
- 4、前端7层负载均衡实现**mobile**、**admin**、**www**站分离。
- 5、设立安全区域, 安全区域只允许**172.16**网段和跳板机访问。
- 6、可尝试实现**JumpServer**跳板机项目。
- 7、搭建**Mysql**高可用架构, 可选用**MHA**方案。
- 8、监控需要全覆盖无死角, 监控系统、网络、应用、业务数据。(业务可调用**LVS**或者**haproxy**网络连接数来展示)

- 9、设计压测模型，通过压测工具，可评估系统的正常容量、峰值、95线、等常见业务指标。**
- 10、有自动化方案管理，可针对不同业务快速进行扩容。**
- 11、有完善的备份方案，可针对不同配置文件、程序代码、数据库等进行有效备份。**
- 12、对日志进行每日压缩归档处理，从业务、应用、等角度进行处理。**
- 13、给出上线方案、回滚、降级方案，并能通过上线方案，更新程序代码从2.5更新到2.6版本。（实现无缝切换）**