

# 高速发展下的电商运维 环境建设之路

饿了么在高速发展的背景下，运维如何从无到有，实现有效的支撑，

保障多业务线的发展、壮大。

@陈翱翔 2015/12/19

# 自我介绍

## 陈翱翔

目前负责饿了么运维工作 见证了服务器数量从数十台到上千台的发展

经历：曾任职于网易邮件、FreeWheel，核心系统一线运维人员。

# Agenda

饿了么简介

运维团队的早期状况

快速发展与团队调整

展望与计划



2009年，从校园起家的饿了么，网站上线

2011年，年交易额破2千万元。

2013年，年交易规模达到10亿元。

十几台服务器，十几位技术人员

2014年，交易规模达到70亿元。

几十台服务器，一个机房，两百来位技术人员

# 2015年7月份公开的数据

4000万用户基数

260个城市、30万加盟餐厅

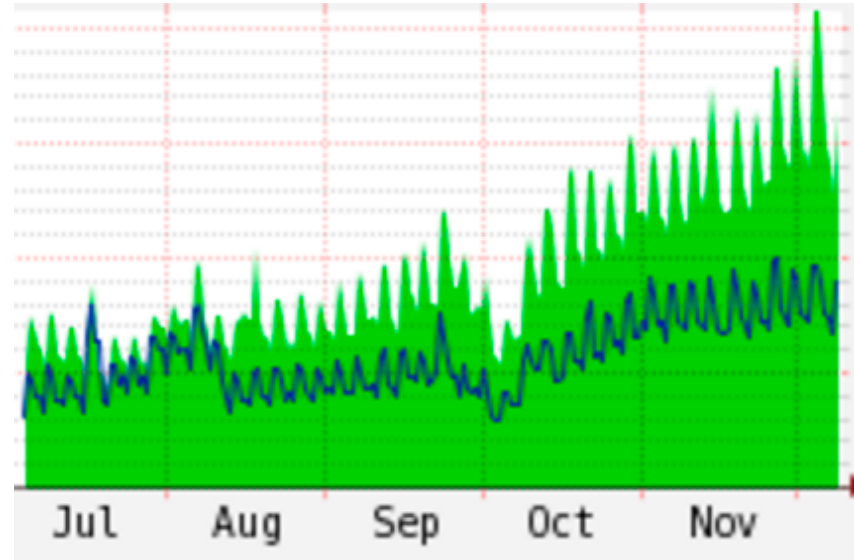
日均交易额破 6000万元



当前：超一千五百台服务器，技术人员超五百人

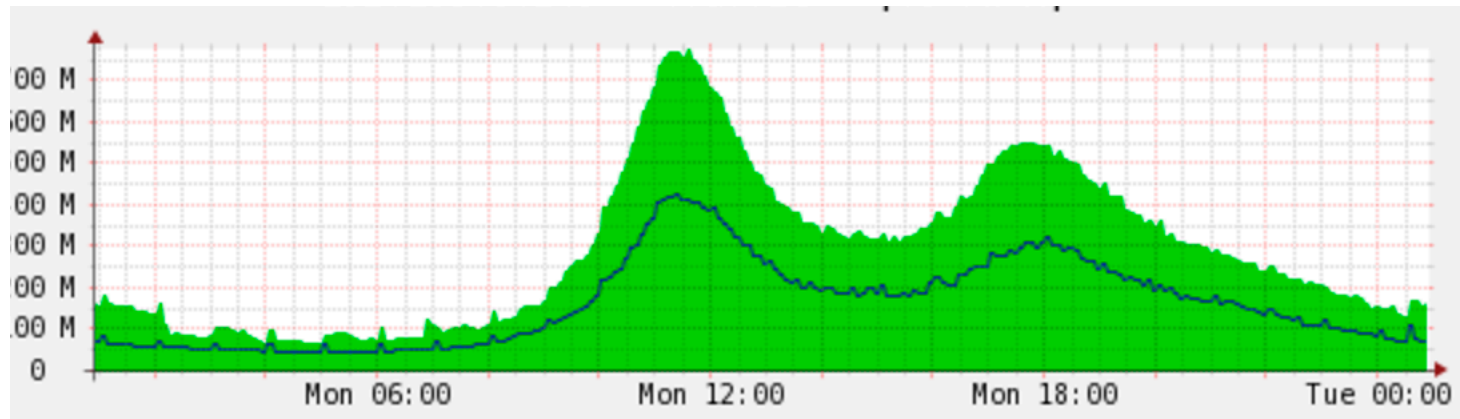
# 业务快速发展

网站流量趋势



百度指数变化

# 一天内的业务流量变化



# 解决最基础的需求

从无到有: 构建基础环境  
核心网规划与建设  
资源选型标准化: 硬件、主机名等  
操作系统、软件、配置标准化



# 基础环境

内网域名系统

机房超前设计

安全控制

虚拟化环境

软负载等基础服务

监控

发布系统

日志分析平台

流程控制

# 基础完善后的一个样例

整改项	改进前	改进后
核心网络	链路从前至后全程单点，排线混乱	双主核心，设计容量约2000台
	无安全防护措施	使用防火墙NAT方式开放服务，隔离非业务端口
	网段、IP地址无规划	生产、办公、测试环境充分考虑。多机房部署考虑
	无远程管理网	完善的管理内网，与业务内网保持对应
	无运维专线及带外管理网	有专线及独立网段，与业务内网隔离，做带外管理用

## 运维工具

- Zabbix + Ztree
- Statsd + Graphite + Grafana
- Jira + Confluence
- ELK
- Saltstack

# 自动化工具 Saltstack

开发、测试、生产等环境  
主机的标准化设置  
应用环境的配置源头  
一年 5000 次代码commit

# 快速发展之下， 业务上的痛点越来越多

产品线多且不同构  
产品内部依赖重，逻辑复杂  
业务变更频繁  
新业务上线需求急

# 人员快速增加，业务线变多

业务驱动、快速试错

产研边开飞机

运维边修飞机

既要高效，又要稳定

需求多10个/天

上线快

上线变更多

人员补充慢

高峰期操作?

业务代码bug?

工具少 手工多

生产事故频发

# 运维人员

运维权责划分、定人定岗  
操作规范、指引  
限制变更  
高峰期待命

# 研发团队

新项目走评审流程

高峰期禁止变更

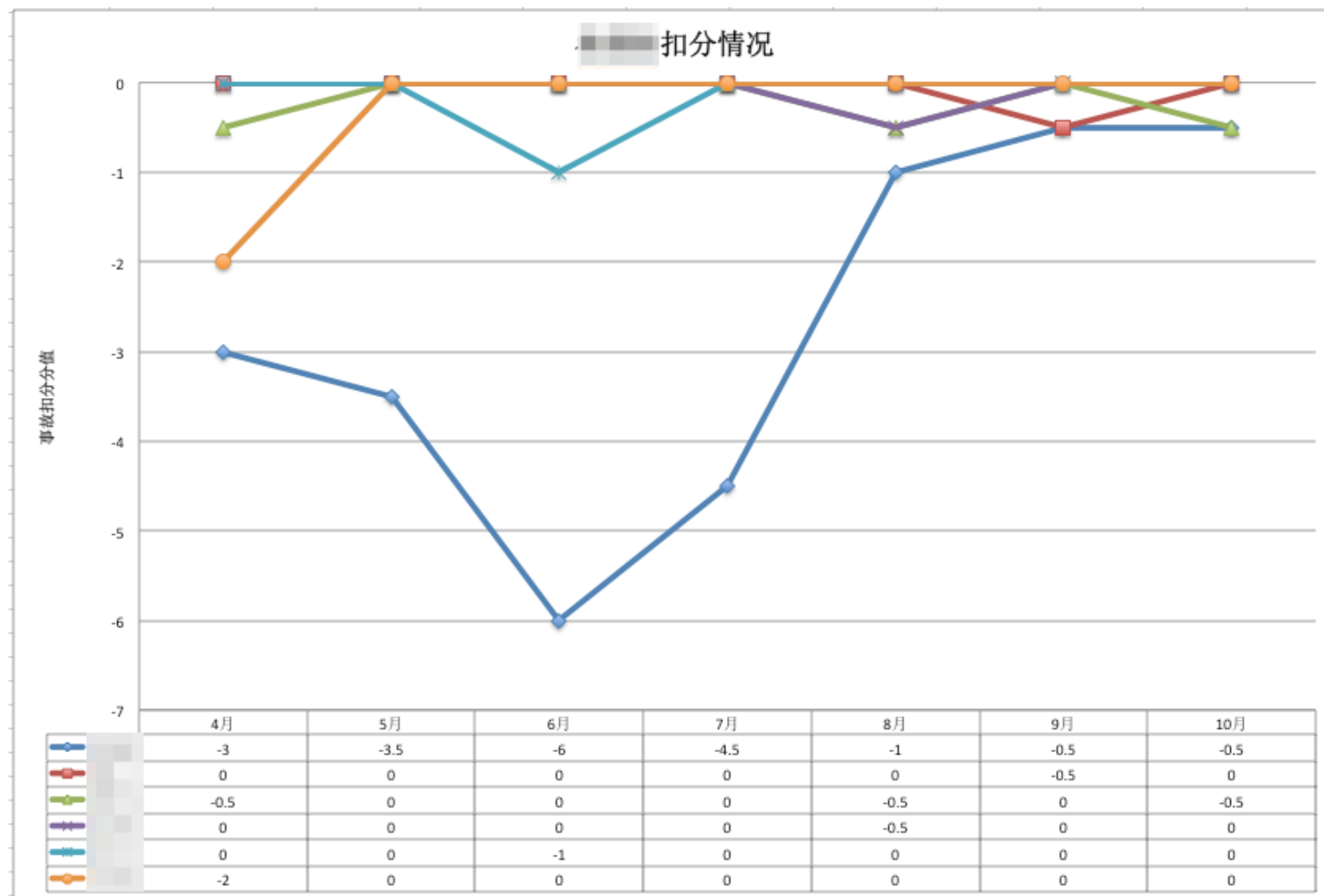
业务与监控对接、SOA等



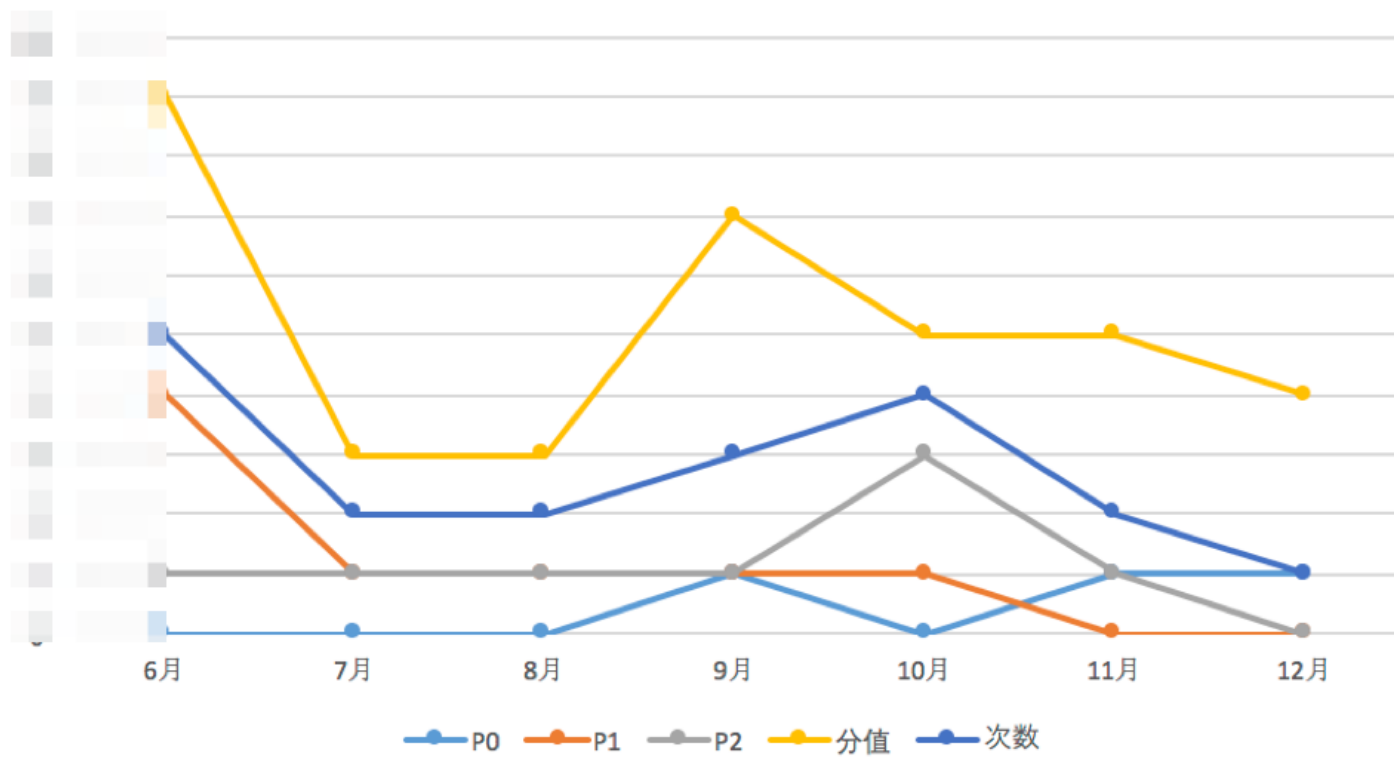
# 技术团队的事件定级及考核规则

事故	P3事故	P2事故	P1事故	
业务 持续 中，	关键业务严重影响，持续>10分钟，≤20分钟；	关键业务严重影响，持续>20分钟，≤30分钟；；	关键业务受严重影响，持续>30分钟，≤1小时；	关键业务 >1小时
业务 严重 持续 分	非关键业务受严重影响持续时间>30分钟，≤1小时	非关键业务受严重影响，持续>1小时，≤22小时	非关键业务受严重影响，持续>2小时	事故时段
	事故时段单量损失≥5%，<10%	事故时段单量损失≥10%，<20%	事故时段单量损失≥20%，<30%	

# KPI之一：技术团队、运维团队故障定级



# 事故次数与积分分布



# Next Steps in 2016:

运维平台化

监控平台的持续完善

Trace用户染色

基础平台智能化（自发现、自管理）

# WE ARE HIRING!

运维工程师@上海：

- Python、Go、Linux

Email: [root@debian.cn](mailto:root@debian.cn)

微信: quebasa

谢谢！

Q & A