

一、基本资料

| | | | | |
|---------|--------|------|---------------------------|-------------|
| 姓名 | 性别 | 出生年份 | 籍贯 | 当前所在地 |
| 吴城 | 男 | 1984 | 浙江杭州 | 北京丰台 |
| 毕业院校 | 专业 | 毕业时间 | 邮箱 | 手机号 |
| 哈尔滨工程大学 | 机械设计制造 | 2008 | jacky.wucheng@foxmail.com | 18612624987 |

二、工作经历

- 2017.11 ~ 至今，量化健康，在IT部担任“技术总监”岗位，负责IT部团队管理和“宏基因组测序分析平台”、“微生物知识图谱系统”的架构设计与核心模块编码工作。
- 2014.10 ~ 2017.04，汽车之家，在系统平台部担任“高级架构师”岗位，负责技术团队管理和“devops平台”的架构设计与核心模块编码工作。
- 2009.01 ~ 2014.09，新浪
 - 在“研发部”先后担任“高级系统开发工程师”和“技术主管”岗位，负责团队管理和“新浪图床系统”、“新浪CDN平台”、“新浪多媒体转码系统”的架构设计与核心模块编码工作。
 - 在“公共服务部”担任“系统管理员”和“PHP开发工程师”岗位，负责“新浪登录注册系统”的运维和后端开发工作，负责“新浪IM在线系统”的架构改造工作。
- 2008.06 ~ 2008.12，网易，在网络系统部担任“IDC系统工程师”岗位，负责IDC服务器和网络设备的上下线、机架部署、网络布线、巡检和故障处理，在云计算数据中心的大规模系统部署和运维方面积累了丰富的经验。

三、专业技能

技术方面

- 熟练掌握Golang/Python/PHP开发，熟悉Java/C/C++的开发
- 熟练掌握Linux下的系统开发和自动化运维的生态知识
- 熟练掌握TCP/IP, DNS, HTTP, Restful等互联网相关技术标准
- 熟练掌握LVS, HAProxy, Nginx等7/4层负载均衡技术
- 熟悉MySQL/PostgreSQL等RDBMS, Redis/Memcached/MongoDB等NOSQL, HDFS/HBase/Ceph等分布式存储, Neo4j/OrientDB等图数据库
- 熟悉Kubernetes/Docker容器技术
- 熟悉大数据生态圈，如Hadoop, Kafka, Spark, ELK
- 具备大规模分布式系统的架构设计和开发经验
- 具备研发项目规划管理能力，具备技术团队管理能力
- 良好的自我驱动能力，积极参与并贡献开源项目

管理方面

- 规划团队长期目标和短期目标，制定绩效考核方法
- 人员招聘，团队组建，对团队人员进行指导培养
- 组织公司内的知识分享，组织跟行业其他公司的线下交流，参加论坛或者大型会议
- 技术选型，方案设计，架构设计，模块和功能设计
- 研发团队的日常管理，包括任务分配、进度跟踪、过程质量控制等等
- 定期项目总结，向下管理，向上汇报

四、项目经历

项目：量化健康宏基因组测序分析平台

项目背景

宏基因组检测业务模式是通过对人体肠道菌群测序分析进而评估健康状况/疾病风险。业务流程是：

1. 用户采样后快递到实验室
2. 样本进入实验室进行SOP流程处理
3. 将DNA送到测序工厂进行测序
4. 对测序工厂返回的基因组数据进行分析得到报告

技术方案

该系统包含了3个重要子系统：客户服务系统、实验室工单系统、workflow计算框架。

该系统采用了前后端分离的开发模式，以Golang为技术栈开发了apiserver封装后端业务逻辑，前端采用React框架实现WebUI。使用“工单模式”实现了实验室SOP流水线，使得实验室操作标准化。

对Argo进行了二次开发实现了workflow自动化，并使得生信团队和IT团队协作解耦。

底层使用kubernetes部署apiserver和各种后台服务，使用MongoDB集群作为数据库，使用GlusterFS作为分布式文件存储。该系统实现了宏基因组检测业务的流水线式自动化。

我的职责

负责技术选型、架构设计、核心组件编码、项目管理和团队管理。

项目：量化健康微生物知识图谱系统

项目背景

肠道菌群被业界称为“人体第二基因组”，目前已知人体肠道包含了2000多种几万株细菌，为了深层探索这些细菌跟人体消化系统、免疫系统、神经系统等方面的相互关系，开发功能菌株和微生态药物，需要一个“知识图谱系统”来做底层的知识支撑。

技术方案

该系统主要包含5个重要模块：爬虫模块、知识抽取模块、数据存储模块、apiserver、WebUI。

1. 用Python开发爬虫从多个公开数据源抓取数据，包括PubMed的医学文献、NCBI的物种分类和基因组数据、BRENDA上的酶数据，KEGG上的代谢通路数据等等。
2. 用Python开发每种数据源的解析抽取程序，将数据写入后端的数据存储模块。
3. 数据存储模块由ElasticSearch和OrientDB组成，ElasticSearch中主要存储医学文献，OrientDB中存储各知识点组成的知识网络。
4. apiserver封装了对数据的各种检索逻辑供客户端调用。
5. 医学顾问和微生物专家则通过WebUI调用apiserver查询知识点。

我的职责

负责技术选型、架构设计、核心组件编码、项目管理和团队管理。

项目：汽车之家devops平台

项目背景

本人所在的系统平台部主要负责公司Infrastructure的开发和运维，但是基础设施环境复杂，人员技能参差不齐，运维体系不规范，程序化运维程度低，整个部门还遭受着“人肉上线代码”、“监控系统”漏报误报、资产盘点屡屡出错，Server和Service资源交付效率低等问题，迫切需要一个devops平台来解决这些问题。

技术方案

该devops平台主要包括了5个子系统：配置管理系统、代码发布系统、资产管理系统、监控系统、资源交付系统。

1. 配置管理系统：基于Puppet ENC架构开发插件实现对常用应用软件的自动化安装和配置管理。可以参考《汽车之家运维团队倾力打造的配置管理系统AutoCMS》。
2. 代码发布系统：基于SaltStack Execution Module二次开发实现了并行命令分发和并行代码上线，接入了公司所有核心业务，彻底杜绝了人肉上线现象。
3. 资产管理系统：基于Puppet Facter开发插件实现了服务器配置参数自动化采集功能，基于“状态机模型”实现了服务器全生命周期自动化管理，用“工单模式”保证了运维操作过程中的“强流程+自动化”，从而保证资产数据准确无误。可以参考《汽车之家CMDB设计思路》，《OpsWorld2016-运维的数据银行》。
4. 监控系统：基于OpenFalcon建设大型分布式监控系统，自研报警故障定位模块和上层产品方案（Golang技术栈），承受住了每秒10万级metric上报的压力，告警及时且能够溯源故障原因。
5. 资源交付系统：用“工单模式”实现需求对接和内部工序流转，且在工单中对接其他子系统API实现后台自动化，现在在小时级别完成资源交付。

我的职责

负责技术选型、架构设计、核心组件编码、项目管理和团队管理。

项目：新浪CDN平台

项目背景

当时公司启动了新浪微博、新浪微盘等新业务，老业务新浪视频的流量也在新业务的带动下不断增长，出于进一步提升产品体验和降低成本的考虑，公司决定自研CDN平台。

技术方案

这是一个规模覆盖全国的基础设施平台，负责静态内容的全局分发，该平台包括3个子系统：全局调度器、L1主干节点/L2边缘节点、OLAP中心。

1. 全局调度器负责用户流量调度和服务器回源流量调度，通过PowerDNS的backend实现了自定义IP库和RoundRobin、WeightedRoundRobin调度算法，支持DNS调度和HTTP重定向调度功能。
2. L1和L2节点由单臂模式的LVS和二次开发的TrafficServer组成，节点集群架构高可用，且规模可横向扩展。整个平台支撑了每秒250G的带宽输出。
3. OLAP中心基于Hadoop技术栈构建，实时汇集边缘节点的WWW日志，定期跑MapReduce任务来分析业务指标。

我的职责

负责技术选型、架构设计、核心组件编码、项目管理和团队管理。

项目：新浪图床系统

项目背景

当时公司启动了新浪微博业务，图片上传和访问量激增，原有的图床系统使用了Netapp Filer存储，不仅价格昂贵，而且因为架构设计有缺陷，使得运维负担很重，集群也不具备扩展性，考虑到未来业务有快速增长的趋势，进行架构改造非常必要。

技术方案

新方案采用了多IDC多活的架构，多IDC间数据使用消息队列进行异步同步。单IDC的集群内使用双HAProxy+多台StoreServer的架构，HAProxy采用一致性Hash算法进行流量调度且相互之间HeartBeat互备，StoreServer（Python开发）不仅负责响应HTTP请求，还负责集群内缓存逻辑。该技术方案不仅实现了单集群规模可横向扩展，扩展时运维工作量极小，集群可靠性也得到了提升，服务器故障也不影响服务，还因为使用了廉价了PCServer替换了Netapp Filer，单集群成本下降了300万。

我的职责

负责技术选型、架构设计、核心组件编码。

项目：新浪多媒体转码系统

项目背景

公司微博业务快速增长，新版本微博从产品定义上要求公司其他业务如新浪视频、新浪微盘上的文件能够在微博上分享和实时播放，且要同时支持PC端和移动端，所以要将种类繁多的媒体格式按照不同的客户端转成相应的目标格式才可以支持该产品需求，这势必需要一个多媒体转码系统来支持。

技术方案

该系统由3部分组成：任务队列、调度器和转码服务器。调度器基于Twisted Python实现，支持FIFO、Priority和抢占式调度模式。转码服务器任务管理程序也由Python开发，底层由FFMPEG实现格式转换。集群规模可以自由扩展，且无单点故障。

我的职责

负责技术选型、架构设计、CodeReview。

五、个人作品

- Blog: <http://jackywu.github.io/about>
- Github: <https://github.com/jackywu>
- 个人共享过的开源项目
 - jackywu/argo: Container-native workflows for Kubernetes.
 - jackywu/bee: Bee is a tool for helping develop with beego app framework.
 - jackywu/salt: Software to automate the management and configuration of any infrastructure.
 - AutohomeCorp/Assets_Report: A assets report plugin for puppet-master based on Factor, https://github.com/AutohomeCorp/Assets_Report.
 - jackywu/vagrant: Vagrant is a tool for building and distributing development environments.
 - jackywu/kraken-extension: utility extension software of kraken.
- 原创技术文章: <http://jackywu.site/categories/technology/>