中等创业公司的后端技术选型

错刀@挖财

http://hongjiang.info

中等互联网创业公司?

怎么定义

- 公司人员规模?
- 产品的用户量/访问量?
- 融资/生存状况?

怎么定义

- 公司人员规模?
  - 研发人员在 50 人以上
- 产品的用户量/访问量?
  - 百万/千万 级别以上
- 融资/生存状况?
  - 已有大致的发展模式

架构就是做选择

- 人的因素
  - 水平、背景、历史因素
- 业务特点
- 生态系统

架构就是做选择

- 扩展性:
  - 可能的变数
- 稳定性
  - 运维的成熟度
  - 经得住考验
- 安全性
  - 是否快速响应
- 没有最优，只有权衡

略微进取一点的保守派

编程语言

- 大部分业务基于JVM生态系统
  - Java8/Scala2.10+
- 少量
  - C++/Go

后端栈

- 负载均衡
- HttpServer
- 应用服务器
- Web/ Front-end层
- 服务调用层
- 缓存
- 持久层

负载均衡/反向代理

- Haproxy
- Nginx/Tengine
- 反向代理与后端应用服务器的连接模式(插件实现)也是需要考虑的

应用服务器

- Tomcat (采用)
- Jetty
- Jboss
- Wildfly/Undertow
  - 不成熟

应用服务器

- Tomcat
  - 对于大多创业公司的访问量BIO足够了，很稳定
  - NIO在7.0.52后的版本也是很稳定的
  - 不建议采用基于native的APR模式
  - 建议单应用部署

参考:

taobao使用ali-tomcat遇到过的一些问题和经验
http://hongjiang.info/index/tomcat/

Web框架

- Spring-MVC（采用)
- Play-framework（内部小范围)
- Struts (不适合)
- Webx (不适合)

Spring

- 依赖注入(DI)是一种协作方式
- 群众基础(非常重要)
- 遵循一种统一的风格
  - 对于团队 xml优于annotation

SOA

- Dubbo
  - 没有其他可对比的(群众基础)

SOA

- Dubbo 的注意问题
  - 默认配置是为阿里的场景预设的
  - 建议线程池采用cached方式
  - 注册信息的本地缓存文件路径设置不要用默认值
  - 写操作不要设置重试
  - Hessian bug很多

链路跟踪

- 淘宝鹰眼 (未开源)
- Twitter zipkin
- 大众点评 Cat

日志采集

- LogStash
- Elasticsearch

消息系统

- Kafka（采用)
- MateQ
- RabbitMQ
- Activemq

消息系统

- Kafka的一些感受和建议
  - 稳定，快速
  - 量不大的情况单分区足够
  - Replica 多份是必须的
  - 注意消息重复的情况
  - 建议发送时启用ack
  - 消费端的offset不建议采用zk维护
  - 合理的设置保留时间

缓存

- Memcached
- Redis (采用)

数据层

- Mysql + Cobar (采用)
- PostgreSql
- Hbase (采用)
- MongoDB (采用)

数据层

- 连接池:
  - Druid (可监控性)
  - 建议在应用上配置，不建议在servlet容器层配置

- 框架:
  - Mybatis
    - 注意类型参数

数据流处理

- Akka
- Akka Streamings
- 其他streaming框架

数据流处理

- Akka
  - Push or Pull ?
  - Back pressure
  - 持久化特性不够稳定
  - 设置合适的Supervisor
  - 与Spring 的整合

模块化

- 依赖冲突与模块化
  - OSGI 或 自实现一个轻量级的隔离?
  - 暂通过脚本自行解决
  - Maven的仲裁机制的问题
  - 未来随着业务和人员的扩张，模块化问题是绕不过的

监控&诊断

- JMX
- 工具&脚本
- Agent
  - 不要因为Agent在每台机器都有而基于它做太多的事情(进程被传染的案例)

安全

- app端通讯走ssl是必要的
- 设置Tomcat的app目录权限只读
- 框架漏洞

安全

- Tomcat的安全隐患 (及时关注，及时更新)

  – eg: 7.0.53以下版本的Dos漏洞
  构造特殊大小的chunked请求

  $ echo -n $'POST /test.war/index.html HTTP/1.1\r\nHost: any\r\nTransfer-encoding: chunked\r\n\r\n1f2f3f4f5f\r\na\r\n0\r\n\r\n' | nc localhost 8080

  注，如果前端用Tengine 不受影响。
  1.2.9版本及以前不支持chunked。
  1.4.X版本支持，但是两种情况:
  一种是收完上传内容，再发往后端，这种会使用Content-Length，没问题。
  一种是收一段转发一段，这种方式tengine（nginx）会重新构造chunk包再发往后端，所以chunk大小可控。

其他

- 序列化
  - 1) json
  - 2) msgpack
  - 3) hessian

随着服务规模膨胀，cpu耗费在序列化/反序列化的比重增加

其他

- 分布式锁:
  - 1) zookeeper
  - 2) redis

其他

- 异步化
- Servlet3.0/3.1
- 静态资源托管以及CDN的选择

其他 scala tips

- Scala编译选项:

  -feature
  -deprecation
  -unchecked
  -explaintypes
  -Yno-adapted-args
  -Ywarn-dead-code
  -Ywarn-unused
  -Ywarn-unused-import

其他 scala tips

- Scala 的陷阱
  - http://hongjiang.info/scala/
  - pitfalls 系列

架构是衍化出来的，不是规划出来的

- 可用优于完美
- 整体优于局部