服务端技术选型

發表於 2015-04-17

**服务框架**

**MVC Framework**：Rose 框架简单易用，并且我米内部服务和工具都优先支持 Rose 项目，默认使用 Rose 框架是很好的选择。文艺一点想做个异步化 web 服务，可以选择 Spring MVC 3.2 以上版本，并搭配高版本 Resin/Jetty 服务器，该方案已有线上服务使用，只是搭项目时会稍复杂些。

**RPC**：我们线上使用的是 Apache Thrift，是 0.5.0 版本。我们计划下一步升级并完善他。具体可以访问我们的 Thrift RoadMap 页面。

**Javascript Library**：随大流用 JQuery，也可以申请前端支持。

**CSS Framework**：用 Bootstrap。

**Database**

**ORM Framwork**：自然是 paoding-rose-jade，成熟好用，无论是简单的数据表操作，或者是“高大上”的分表分库都不在话下。更集成了 perfcouter 功能，可以实时查看和监控 MySQL 语句执行效率。

**数据库连接池**：jade 使用 Apache DBCP。Tomcat JDBC 声称更快更强，还没测试过，暂时还是继续 Apache DBCP 吧。

**No SQL**：Hadoop, Hive, HBase。

**Cache**：本地 cache 使用 Ehcache，Guava 的 cache 也可以使用。中央式缓存，用 Memcached。

**传统数据库**：MySQL。线上业务先和 DBA 确认服务器磁盘是否是 SSD。

**Utilizes**

**General**：Apache Commons 必备。必须使用 3.0 以上版本。Guava 是 Google 推出的产品，新鲜的功能更多一些。

**JSON**：[Jackon](https://github.com/FasterXML/jackson-databind/" \t "_blank)功能强大。

**XML**：据说 JDK 自带的 JAXB 就很好。

**Email**：直接用 Spring 的封装。

**Logging**：Slf4j + Log4j + Scribe + Kafka。新同学掌握 Slf4j 和 Log4j 就好。

**Schedule**：使用 Spring 的 Schedule。Spring Cook Book 部分有演示代码。

**时间日期**：JodaTime。请看我们的代码示例自学。

**io**：简单的文件 io 用 apache commons 或者 google guava。zip 文件处理用 zip4j。

**读取命令行参数**：[JCommander](http://jcommander.org/)是一个简单好用的命令行参数解析框架。

**Test**

**Spring测试框架**：Spring Test 配合 JUnit 非常顺畅，需要确认项目中使用 Spring 版本 3.0 以上版本。

**Mock**：Mockito 是现在最优雅简洁的 mock 框架了，强烈推荐使用。Mockito 搞不定的部分，比如static 函数，搭配 PowerMock。

**数据库测试**：使用 H2 Database 内存数据库。还是怕慢？用 maven-surefire-plugin 多线程执行测试任务。

**功能测试**：在 onebox 环境测试，一遍遍发布部署太麻烦，使用 jetty-maven-plugin，一键就可以把服务启动起来。

**Performance/Stability Test**：Jmeter 是成熟的工具。

**Development Environment**

**JDK**：线上大多数项目使用 JDK6，JDK8 版本会从离线服务开始用，Maven 编译的包需要保证 JDK6 源码与二进制兼容。

**版本控制**：用 Git，先阅读我们的 Git 文档。如果想深入了解Git，推荐阅读《Git权威指南》。

**构建工具**：用 Maven。需要熟练掌握以下技巧：1.使用 Maven 打包，比如 Jar、War、Jar with dependency、distribution 包；2.使用 Maven 发布包；3.熟知 Maven依赖管理原理。推荐阅读《Maven权威指南》。

**应用服务器**：Nginx + Resin，Nginx 作为反向代理，Resin 承担应用服务器和 Java 容器的角色。需要注意，线上服务记得要调优 JVM，线程数等参数。测试环境用 Jetty 插件就好。