

开工第一天，深度剖析公司技术栈！

王森 java进阶架构师 今天

点击上方“[java进阶架构师](#)”，选择右上角“置顶公众号”

20大进阶架构专题每日送达

进阶架构精品专题

- [Mysql优化专题 \(★★★★\)](#)
- [网络协议专题 \(★★★★\)](#)
- [其余18大专题，请在主页菜单栏查看](#)
- **后台回复【加群】，获取BAT真实面经**



目的

年后伊始，主要是想从架构的角度剖析一下公司使用的一些技术栈都有哪些，都解决了哪些问题，以及几十人团队的日常和项目管理的方式方法。不是说他比别的团队优秀，只是我自己想记录一下做到心中有一个全貌，如果能给你们一点启发我会更高兴。

铭记

不以解决实际问题引入的技术都要流氓，下面我们就来看看都有哪些技术栈。

前端技术栈

1. 使用css3作为页面的样式控制

解决页面的渲染问题

2. 使用Html5作为页面结构的标签语言

解决移动端的页面兼容性问题

3. 使用seajs框架

来解决项目中模块化开发和加载的问题

4. 使用Jquery前端js框架

解决方便操作页面元素的问题

5. 引用流行的前端插件实现业务需求

- 使用bootstrapValidator作为表单的验证插件
- 使用bootstrap进行页面的布局
- 使用layer作为弹窗插件
- 使用jQuery cxSelect 实现下拉列表的多级联动
- 使用bootstrap fileinput完成多文件的上传
- 使用icheck完成对多选框的美化
- 使用百度ueditor解决复杂内容编辑场景

6. 自定义业务插件抽取

主要是解决系统中导出使用的功能代码重复性问题。

- 收藏功能
- 点赞功能
- 评论功能
- 留言功能
- 关注功能
- 页面多处个人信息的设置

- 页面多处商品页面调起支付的功能
- 题库的组卷系统中的试题栏

后端技术栈

1. 项目分层思想实现分布式架构

core层：与业务无关，提供基础的能力，比如：数据库持久，redis缓存,http封装，通用工具。

统一的core项目，提供基础的能力

- we-core-Db项目：mybatis的mapper的基础上做了二次封装
- we-core-redis项目：基于jedis的基础上做了二次封装
- we-core-web
 - 存放一些与web环境下使用的工具类和处理器
 - 自定义web环境里使用的aop注解
 - 统一的异常处理器
 - 自定义分页标签的


base层：该层中的项目有且只能代表一个真实存在而且能独立存在的核心实体对应的业务。

business层：业务层，解决多端，多项目公用的业务流程

web层：为互联网用户提供对外服务，在这层的每一个项目都有自己不被共享的业务。

2. 抽取独立服务实现服务共享

- 好多项目都会使用到很多相同的服务，我们会把它抽取成独立的项目，先以jar包的方式提供公用。
- 比如：短信服务，邮件服务，字典服务，收藏服务，关注服务，订单服务，支付服务

>  we-base-cms [we-base-cms master]	45
>  we-base-dictionary [we-base-dictionary master]	46
>  we-base-follow [we-base-follow master]	47
>  we-base-goods [we-base-goods master]	48
>  we-base-leavemessage [we-base-leavemessage master]	49
>  we-base-order [we-base-order master]	50
>  we-base-sms [we-base-sms master]	51
>  we-base-user [we-base-user master]	52

3. 基于jedis的基础上做了二次封装，实现java对redis的读写

应用场景：

- 整个项目使用统一的id生成策略
- 使用缓存进行存放freemark代码片段的内容，解决高频查询数据给服务器带来的压力

4. 使用maven进行项目的管理

- 使用maven自定义骨架可以实现公司内部创建项目模板化，减少大家创建项目时的时间，创建完就能使用
- 使用maven的聚合的特性实现多项目统一进行构建
- 使用maven继承的特性实现同一项目的不同maven工程的相同配置问题
- 使用maven的属性我们可以使用自定义变量以及访问其他配置项值解决我们配置重复的问题
- 使用maven帮助我们规范工程资源的的存放
- 使用maven的依赖特性可以解决我们，jar包之间的依赖，排出依赖。
- 使用maven仓库的http代理可以实现让maven帮助我们查找，下载所需的jar包，我们只需要指定依赖的gav
- 使用maven我们把公司自己的开发的项目发布到nexus上供其他的项目使用

5. 使用nexus进行内网私服搭建

- 解决jar包重复下载的问题
- 解决公司内部项目共享的问题
- maven仓库以外的jar包管理

6. 使用gitlab进行项目的版本管理

- 解决团队多人协同开发项目的问题，我们团队制定了使用git开发的规范流程。

7. 使用jenkins/hudson实现项目的自动化部署

- 解决项目自动化构建，发布，回滚等问题
 - 定时构建及时发现项目中的问题
 - 实现一键发布缩短发布的时间
- 自动构建和发布的大致流程：
- 从gitlab上拉取最新的代码
 - 使用maven进行构建项目
 - 将构建完成的项目使用scp复制到指定服务器中
 - 并通过远程调用指定服务器的脚本进行项目的发布

8. 使用mysql进行数据存储

- 这个不用多解释。

9. 使用Spring+SpringMvc+Mybitas

- 使用经典的ssm作为平台的MVC框架, 处理http请求,处理业务逻辑，访问数据库。

10. 使用hibernateValidator

- 实现后端接收到的参数进行参数的合法性验证，像非空，最小，最大，范围等

11. 使用spring mvc的拦截器

- 实现项目内的权限控制，比如：sso权限验证，角色的验证，支付的验证等

12. 使用spring mvc的AOP

- 实现多处相同代码抽取出来使用AOP的方式进行公用，比如：json视图的固定格式返回

13. 使用logback

- 作为架构的日志框架

14. 使用flying-saucer+itext+freemark

- 解决项目中pdf的应用场景，比如：学生下载的准考证，测试报告等。

15. 使用JavaMail技术

- 实现邮件的发送服务

16. 使用TestNg+H2

- 实现进行单元化测试，保证代码质量

17. 使用Cookie+Redis

- 实现平台的单点登录

18. 实现SpringMvc提供的HandlerExceptionHandlerResolver接口

- 使用spring提供HandlerExceptionHandlerResolver实现全站统一异常处理
包括：业务异常，登录超时异常，其他异常。

19. 实现SpringMvc提供的ServletContextAware接口

可以实现项目启动时添加一些操作。

- 初始化freemarker的配置
- 初始化项目全局共享的静态变量（域名，文件存储目录等）

20. 使用Quartz

- 实现定时任务的处理

21. 使用网络爬虫技术Jsoup

- 解决爬取其他网站的数据信息问题

22. 使用Qrcode

- 解决生成二维码的功能

23. 使用jsp的自定义标签

- 解决全站分页的实现

24. 使用阿里的Druid

- 作为连接数据库的连接池

25. 使用spring的ioc容器解决对象的依赖注入

- 使用ioc实现指定特定接口的实现类以应对业务实现的变化
- 使用ioc实现业务中需要的配置注入，防止参数值写死

26. 使用CXF或者Axis

- 实现通过WebService和第三方进行对接

27. 结合自定义注解+反射

- 实现对方法访问的控制以及添加额外业务逻辑
- 比如:个人信息在多个请求Controller中都需要填充

28. 使用Mybatis框架

- 解决数据持久化的问题
- 引入开源的: Mybatis的Mapper接口的方式实现支持单表的curd, 避免写更多的重复代码
- 引入pagehelper解决mybatis的分页问题
- 使用Mybatis的Interceptor实现sql的打印功能

29. 使用Jsp,Freemark

- 作为视图层的处理技术

30. 使用Freemark+字符串替换实现自定义vt标签

- 自定义视图标签主要为了解决像字典那种字段不断需要跨表查询的问题。

31. 自定义json视图

- 自定义json视图是为了实现规定json返回的统一格式。

32. 使用El表达式和jstl标签库

- 解决访问model中的数据以及页面上显示的逻辑处理

33. 使用自定义注解实现某些功能的标注

- 在处理json返回值的时候可以在标注一下@Void就表示不需要包裹
- 在sso拦截权限的时候可以标注一下方法是否需要拦截
- 在多个Controller里都需要设置用户信息到页面的时候, 我们可以利用Aop+Annotation的方式实现

服务器环境

- 内网测试环境与线上运行环境统一更多保证上线成功
系统版本
软件版本
使用脚本一件安装

- 使用nginx做代理服务
实现动静分离
实现ip访问的限制
根据http的请求信息实现转发其他请求
- 使用redis做业务缓存
- 使用tomcat做web容器
- 使用JDK8作为jdk版本
- 使用mysql作为数据库
- 使用mount的方式做静态资源的挂载

团队管理篇

管理的理念：倡导自主管理，学习，进步，开放，分享

1. 推行java开发规范统一团队编码风格

解决人员流动带来的业务梳理的难度，提升同志们的编码水平，做一个有追求coder，大致包括以下几个部分：

- 项目命名规范
- 项目产出标准化规范
- 开发规范
- 前后端接口规范
- java编码规范
- url定义规范

2. 推行数据库操作规范，保证数据库的整洁

- 禁用存储过程
- 命名规范
- 字段类型
- 每次由于业务的变化需要升级sql语句，需要保存记录，在内网测试完成才可以线上执行

3. 使用Workdtile作为团队的项目，任务管理工具

- 项目的资料，产出，会议，重要的时间节点，分享都放在这上面
- 可以让项目组成员更加注重自己的时间管理，任务管理，提高计划的能力
- 会培训如何写计划，写计划是否可行，是否有问题都会有兄弟们给你指出

4. 搭建内网wiki让团队的各种产出都固化起来

解决团队文档多人编辑，以及知识的固化。

大概的目录：



5. 进行结对编程

- 有新人补充为了快速上手业务和熟悉技术架构，采用老人带新人进行一对一的结对编程，既可以让新人快速融进这个团队，也可以让老人对自己的知识再一次进行加深和固化。

6. 提倡和要求读书

- 团队新人会推荐一波入伙必读书籍：

入伙推荐书单

穷人在怀疑中拒绝，富人在怀疑中了解

这些书里有团队的工作方式和学习的习惯。

这些书有团队的做事风格和价值观。

书单推荐顺序：

软能力

没有实实在在的技术可以对应，但是对生活和工作却有很重要的影响

- 《小强升职记》---本书向你介绍了让工作变得更加轻松自如、事半功倍的方法。
- 《把时间当朋友》---如何打开心智，运用心智开启自己的人生成功之旅。
- 《如何阅读一本书》---本书主要论述指导如何通过阅读增进理解力。
- 《如何高效阅读》---教你如何让你的阅读速度提升10倍以上。
- 《一万小时天才》---1万小时的锤炼是任何人从平凡变成超凡的必要条件。
- 《人性的弱点》---讲述了日常生活和社会交往中的各种技巧，将人际关系提升到一门艺术的高度。
- 《程序员修炼之道》---程序员的内功心法

硬能力

就是实实在在有对应的技术对应的能力

- 《代码大全》---让你了解软件构建的流程及注意事项，成为一个给力的项目负责人
- 《代码整洁之道》---写出更优雅的代码
- 《重构》---写出让人类看得懂的代码
- 《UML基础、应用和案例》---需求分析和设计的利器

- 要求每天早上留出时间去读书，并每周安排人做每周一书的分享

7. 编码习惯要求

上午要求对自己要做的事情进行详细规划，并写出具体的实施步骤写到xmind上，和别人进行沟通验证自己的思路是否正确，下午进行编码实现。

8. 项目一周定期至少两天进行成果演示

- 对于做的快和好的同学进行鼓励，刷存在感
- 对于做的慢的同学，帮助他解决问题，找出原因，指定擅长的人进行解决
- 对于整个产品而言可以及时验证是否符合业务的需求，对完整性进行要求

9. 项目开发周期内对项目的主要成员进行相互的CodeReview

- 根据团队的开发规范去验证编码是否存在问题
既可以熟悉规范，也可以起到大家编码行为一致。
- 传递好的编码习惯和风格
- 讨论和验证业务的编码实现是否正确和合理

10. 每日工作计划

- 说清楚今天做什么，预计达到什么目的
 - 有没有遇到什么困难，需要谁的帮助
 - 有没有个人的原因的潜在风险
 - 是否需要协调资源
- 这个是我们具体的实施方案：每日工作计划

软件构建篇（项目管理）

这里的软件构建指的是做项目的整个流程和步骤，这是我们一起总结出来的。

1. 项目立项

- 保证项目的难度系数在可控范围内。
- 确定项目的量级能够符合实际的项目周期。
- 确定项目的周期（起止点）。
- 确定项目需求的终稿，明确做的业务边界。
- 准备资源做项目（人）。
- 创建worktile项目。
- 将项目资源进行归放到项目标准文档产出。
- 做出让需求方认可的project。
- 召开项目立项会议。

2. 需求分析

- 让产品讲解项目信息以及原型，了解项目背景，这个项目要解决的是什么问题
- 根据项目原型梳理出系统的流程图

- 拿着流程图和产品进行讲述，确认流程理解一致
- 抽取项目中包含领域知识的名词列表，使用一句话能够描述清楚
- 与产品进行沟通，将自己的描述和产品进行check，保证描述的是正确的
- 抽取项目的用例，写到worktile项目中，确保能够覆盖全部需求
- 将用例进行分配人，标时间，拆用例，排顺序以便严格按照用例进行开发。
- 具有复杂的业务，算法的用例和产品进行确认，保证用例描述了产品的需求

3. 系统设计

- 对名词列表进行去重，合并，归类处理，为抽取类图作准备。
- 根据上一步处理后的名词列表画出系统的类图（类，属性，注释）。
- 根据产品原型验证原型中出现的名词是否都已经覆盖（查漏补缺）。
- 根据产品原型中的需求，流程确定类与类之间的关系，画出类的关系图。
- 根据产品的需求分析，抽取出对象的方法和伪代码。【暂时挂起】
- 对复杂的业务进行系统设计。
- 基于项目的分层对系统的服务进行划分部署。
- 根据产品原型中跳转的页面，对URL进行规划，出一份完整的URL列表。

4. 产品研发

- 根据分层服务部署图创建base, bussiness,web层eclipse项目。
- 创建base, bussiness,web层中对应的实体，注释，service。
- 创建数据库，使用建表的程序生成数据库表格。
- 配置nginx，配置host,配置jetty保证能够使用域名访问成功。
- 根据编码规范参照URL列表将项目中对应的物理文件创建出来。
- 参照项目用例的实现步骤使用编程语言翻译成代码敲出来。
- 每天对编写的代码进行codeReview，保证代码符合规范。
- 在本地进行充分的自测，保证每一个用例是完整正确的。
- 每天定点进行团队内部演示已经开发完的用例，记录发现的问题，保证用例完整

5. 产品测试

- 合并代码到master上。
- 添加项目到自动化集成项目中去。
- 同步一份外网全新的数据到内网数据库中，保证和线上的数据环境一致。
- 在内网数据库里执行sql的升级日志。
- 修改内网服务器配置（nginx,tomcat等）。

- 通知业务，需求方进行项目的内网测试，并修改他们发现的bug。
- 测试业务的流程（只关注流程），修改测出的bug。
- 根据项目的用例测试每一个功能，修改测出的bug。
- 测试页面的兼容，交互，样式等，修改测出的bug。
- 没有影响用户使用的重大bug前提下，并且业务，需求方认可就可以发布上线。
- 解析域名到代理服务器上。
- 重复执行4-9的步骤
- 得到需求和业务方的肯定（交付）之后，并且系统稳定运行一段时间之后，项目宣布结项。

6. 项目总结

- 按照项目的标准产出补齐缺失的文档。
- 每一个人使用xmind列出在这个项目中的收获。
- 项目负责人组织大家一起讨论每个人的收获列表，整理出可以补充完善团队流程和技术的列表。
- 根据上一步的列表，安排上人，排上时间进行固化，总结，在团队内进行推广。
- 每一个人使用xmind列出你在做的过程中耗时长，复杂度高，重复性高的地方。
- 项目负责人组织大家一起整理出是由团队的机制，流程，技术等不足导致的问题列表。
- 根据上一步的列表，为每一个问题安排上负责人以项目的方式解决这个问题。
- 邀产品（需求方）演示项目的主要用例和主流程，确认无误后方可进行内测阶段。

团队使用工具

1. 使用Eclipse和Idea

作为团队开发的IDE,前期使用的Eclipse作为团队的IDE，后期为了解决eclipse运行慢和不断出现的异常，也为了迎合潮流决定转投Idea，不管是什么IDE我们都强调统一：

- 统一的版本
- 统一个插件配置

- 统一的特性设置
字体
提示
代码模板
.....
- 必须要掌握的快捷键，全队都得熟悉

2. 使用花生壳或者ngrok

实现内网穿透技术，解决微信支付或者接口的调试工作。

3. 推荐团队使用的其他工具

- 使用xmind作为思维导图工具
- 使用Everything作为文件的搜索文具
- 使用为知笔记作为知识管理工具
- 使用Notepad++作为高级编辑器
- 使用Postman作为后台接口测试工具
- 使用SecureCRT作为连接linux服务器的工具
- 使用navicat作为连接数据库的客户端工具
- 使用jetty作为web开发的容器
- 使用SourceTree作为Git客户端管理工具
- 使用Powerdesigner作为建模工具
- 使用Visio作为流程图制作工具
- 使用lantern作为vpn工具
- 使用Teamviews作为电脑远程控制的工具
- 使用jd-gui作为java反编译的工具
- 使用ffmpeg作为视频处理的工具

- 使用Snipaste作为截图工具
- 使用屏幕录像专家作为录屏工具

原文：www.cnblogs.com/wangsen/p/9329735.html

你们公司呢？用的什么技术渣？评论区分享一波



看到这里，说明你喜欢本文
你的转发，好看的人都点了好看↓↓↓

阅读 1760

好看 22

精选留言

写留言



Just Do It

3

你这技术栈的好多技术偏老啊。



廖锐

写的很好，学习了。多问一句，作者是哪家公司？

作者

2

是不是准备过来面试就按上面技术栈复习了



殊麒

1

技术陈旧传统，估计是软件类公司而非互联网公司



Jw

1

运维的技术不写？



管住自己的嘴

已经很全面了



Fight Song

把这些会了，能月薪1万么👉



支付宝

额



皖小生

受教



白立武

666回家吃饭，回头仔细阅读

作者

放心不会再看的了--



catch

最后一句，技术栈打错了。。。

作者

故意故意



permanence

写的详细些，不过框写的挺好的



橙诺

你讲公司的技术栈，你不怕被你公司开除吗

作者

放心，老板不看技术文的



小小

前端不用三大框架么

作者

额....
