目录

[1 Java-开发部署 3](#_Toc11043)

[1.1 工具辅助 3](#_Toc21436)

[1.1.1 开发集成 3](#_Toc29380)

[1.1.2 版本管理（VCS） 4](#_Toc31711)

[1.1.3 建模管理 5](#_Toc23527)

[1.1.4 项目管理 6](#_Toc32759)

[1.1.5 架构图管理 6](#_Toc25790)

[1.2 开发要求 6](#_Toc11061)

[1.2.1 开发规范 6](#_Toc600)

[1.2.2 设计模式 7](#_Toc8842)

[1.2.3 代码重构 7](#_Toc8600)

[1.2.4 代码整洁 7](#_Toc21553)

[1.2.5 代码大全 7](#_Toc265)

[1.2.6 代码模板 7](#_Toc28873)

[1.3 部署辅助 7](#_Toc12838)

[1.3.1 Linux 7](#_Toc8678)

[1.3.2 脚本 8](#_Toc17935)

[1.3.3 CICD（持续集成开发部署）和Devops 9](#_Toc4741)

[2 Java-扎实基础 11](#_Toc2246)

[2.1 Java核心技术 11](#_Toc16327)

[2.1.1 Java核心基础 11](#_Toc1893)

[2.1.2 JavaAPI 11](#_Toc31965)

[2.1.3 Java源码 11](#_Toc16617)

[2.1.4 各版本jdk新特性 12](#_Toc11030)

[2.2 GUC 12](#_Toc7823)

[2.3 网络与IO 12](#_Toc28504)

[2.3.1 网络协议 12](#_Toc30672)

[2.3.2 IO 12](#_Toc14495)

[2.3.3 NIO 12](#_Toc18302)

[2.3.4 网络编程（socket/serverSocket） 13](#_Toc18225)

[2.4 JVM 13](#_Toc8852)

[2.5 数据结构与算法 13](#_Toc8869)

[3 Java-亿级流量技术栈 14](#_Toc18100)

[3.1 关系型数据库(不用过分深入到dba) 14](#_Toc7410)

[3.1.1 SQL基础与高级使用 14](#_Toc1385)

[3.1.2 数据库分类 14](#_Toc13085)

[3.2 JavaWeb 15](#_Toc15016)

[3.2.1 前端基础 15](#_Toc24644)

[3.2.2 前端技术栈 16](#_Toc12323)

[3.2.3 javaweb核心基础 18](#_Toc24090)

[3.2.4 web容器 19](#_Toc7276)

[3.2.5 混乱的java日志体系(排查问题) 19](#_Toc26118)

[3.3 JavaEE 20](#_Toc9210)

[3.3.1 MVC框架 20](#_Toc24713)

[3.3.2 模板引擎（至少会1-2种） 21](#_Toc13197)

[3.3.3 Orm框架 21](#_Toc7324)

[3.3.4 Spring 22](#_Toc1674)

[3.4 专题框架(针对某一个领域的框架) 22](#_Toc14110)

[3.4.1 Webservice(慢慢过时) 22](#_Toc315)

[3.4.2 Http 22](#_Toc23178)

[3.4.3 Xml 23](#_Toc1566)

[3.4.4 Json 24](#_Toc398)

[3.4.5 Excel操作 24](#_Toc26281)

[3.4.6 工作流 25](#_Toc31553)

[3.4.7 缓存 25](#_Toc21293)

[3.4.8 权限与安全 26](#_Toc9727)

[3.4.9 定时任务 27](#_Toc2743)

[3.4.10 单元测试 27](#_Toc15928)

[3.4.11 Nosql 27](#_Toc21285)

[3.4.12 数据库连接池 28](#_Toc18620)

[3.4.13 文档与调试 28](#_Toc17103)

[3.5 微服务SpringBoot 28](#_Toc7231)

[3.6 分布式架构 29](#_Toc28776)

[3.6.1 主流分布式架构 29](#_Toc27694)

[3.6.2 分布式缓存 32](#_Toc780)

[3.6.3 全文搜索引擎（大数据量mysql再怎么优化都不行） 32](#_Toc1809)

[3.6.4 elastic stack (el栈) es+logstash+kibana(ELK)现在慢慢被EFK替代，F就是beat 32](#_Toc11156)

[3.6.5 分库分表 33](#_Toc19720)

[3.6.6 分布式任务 33](#_Toc6562)

[3.6.7 分布式事务 33](#_Toc18216)

[3.6.8 消息队列mq 34](#_Toc21000)

[3.6.9 分布式链路追踪 35](#_Toc20161)

[3.6.10 分布式锁 35](#_Toc14714)

[3.6.11 分布式对象存储(图片文档js等) 36](#_Toc27605)

[3.6.12 负载均衡 36](#_Toc32164)

[3.7 三方源码 37](#_Toc3143)

[3.8 测试技术 37](#_Toc6172)

# 1 Java-开发部署

## 1.1 工具辅助

### 1.1.1 开发集成

#### 1.1.1.1 Eclipse

|  |
| --- |
|  |

#### 1.1.1.2 Idea

|  |
| --- |
|  |

#### 1.1.1.3 Myeclipse

|  |
| --- |
|  |

#### **1.1.1.4 Vscode**

|  |
| --- |
|  |

#### 1.1.1.5 Wechat开发

|  |
| --- |
|  |

### 1.1.2 版本管理（VCS）

#### 1.1.2.1 svn系列

##### 1.1.2.1.1 Subversion

|  |
| --- |
|  |

##### 1.1.2.1.2 TortoiseSVN

|  |
| --- |
|  |

#### 1.1.2.2 Git系列

##### 1.1.2.2 .1 Git

##### 1.1.2.2 .2 Github

##### 1.1.2.2 .3 Gitlab

##### 1.1.2.2 .4 Gitee

### 1.1.3 建模管理

#### 1.1.3.1 UML建模

##### 1.1.3.1.1 UML

类之间的关系组合，聚合，继承，实现，依赖

用例图，类图，时序图

##### 1.1.3.1.2 常用作图工具

###### 1.1.3.1.2.1 Staruml

###### 1.1.3.1.2.1 RationRose

#### 1.1.3.2 数据库建模

##### 1.3.2.1 PowerDesigner

### 1.1.4 项目管理

#### 1.1.4.1 Maven

◎[1-Java-开发部署\1-工具辅助\4-项目管理\1-Maven\maven.xmind](1-Java-开发部署/1-工具辅助/4-项目管理/1-Maven/maven.xmind)

#### 1.1.4.2 Gradle

maven作用一样，spring源码已经从maven改成gradle

#### 1.1.4.3 Ant

### 1.1.5 架构图管理

## 1.2 开发要求

### 1.2.1 开发规范

### 1.2.2 设计模式

◎ [1-Java-开发部署\2-开发要求\2-设计模式\设计模式.xmind](1-Java-开发部署/2-开发要求/2-设计模式/设计模式.xmind)

### 1.2.3 代码重构

◎[1-Java-开发部署\2-开发要求\3-代码重构\代码重构.xmind](1-Java-开发部署/2-开发要求/3-代码重构/代码重构.xmind)

### 1.2.4 代码整洁

◎[1-Java-开发部署\2-开发要求\4-代码整洁\代码整洁.xmind](1-Java-开发部署/2-开发要求/4-代码整洁/代码整洁.xmind)

### 1.2.5 代码大全

### 1.2.6 代码模板

## 1.3 部署辅助

### 1.3.1 Linux

#### 1.3.1.1 基本命令使用

◎[1-Java-开发部署\3-部署辅助\1-Linux\Linux.xmind](1-Java-开发部署/3-部署辅助/1-Linux/Linux.xmind)

#### 1.3.1.2 协议

##### 1.3.1.2.1 NFS

##### 1.3.1.2.2 FTP

##### 1.3.1.2.3 SSH

##### 1.3.1.2.4 SFTP

#### 1.3.1.3 常用软件安装（nginx,jdk...）

#### 1.3.1.4 Centos7用的比较多

### 1.3.2 脚本

#### 1.3.2.1 Shell

#### 1.3.2.2 Python

#### 1.3.2.3 Lua

### 1.3.3 CICD（持续集成开发部署）和Devops

#### 1.3.3.1 容器技术Docker

镜像

容器

数据卷

docker安装常用软件

centos

tomcat

mysql

nginx

es

kibana

redis

docker镜像原理

docker的启动流程

docker网络

镜像仓库

共有

dockerhub

阿里云

私有

docker registry

harbor

#### 1.3.3.2 Kubernetes(K8S)

术语概念

组件

master

api server

controller manager

etcd

scheduler

node

kubelet

kuber proxy

资源术语

pod

controller

service

ingress

集群搭建

kubeadm官服工具

mknikube

二级制方式手动安装

常用工具

dashboard官方web管理界面

kuboard

实战练习

elk日志收集与可视化查询

Prometheus+grafana监控与预警

skywalking分布式链路追踪

#### 1.3.3.3 Jkens(做cicd的)

# 2 Java-扎实基础

## 2.1 Java核心技术

### 2.1.1 Java核心基础

◎[2-Java-扎实基础\1-Java核心技术\1-java核心基础\1-Java核心基础.xmind](2-Java-扎实基础/1-Java核心技术/1-java核心基础/1-Java核心基础.xmind)

面向对象

基础

变量

流程控制

面向过程

成员变量，方法，对象，类，继承，封装，多态，泛型

反射

异常体系

常见API

基本类型包装类

date,simpledatefomat,localdatetime

StringBuffer,StringBuilder,StringBuffer

Math,Random

java常用类与一些工具包

commons,包括apache的commons

guava

集合操作

collection

map

jdbc

jdk8新特性、jdk9..jdk10，16（jdk8后收费）

swing（窗体，有界面）-企业用的不多，初学者喜欢

XML(可扩展标记语言)

### 2.1.2 JavaAPI

### 2.1.3 Java源码

### 2.1.4 各版本jdk新特性

## 2.2 GUC

线程与JUC(并发)-高级必经之路

## 2.3 网络与IO

### 2.3.1 网络协议

#### 2.3.1.1 Http协议

#### 2.3.1.2 TCP/IP协议

### 2.3.2 IO

### 2.3.3 NIO

### 2.3.4 网络编程（socket/serverSocket）

#### 2.3.4.1 Socket

#### 2.3.4.1 ServerSocket

#### 2.3.4.1 Mina

#### 2.3.4.1 Netty

## 2.4 JVM

JMM模型

常见垃圾回收算法

垃圾回收器

JVM调优

JVM内存管理

## 2.5 数据结构与算法

数组

链表

hash

树

排序与查找

冒泡

快排

选择排序

折半查找

递归

# 3 Java-亿级流量技术栈

## 3.1 关系型数据库(不用过分深入到dba)

### 3.1.1 SQL基础与高级使用

1-SQL基础语法

ddl,dml,dql

子查询

join,union,group

视图

触发器

存储过程

2-SQL高级技术

索引结构(面试最多)

索引分类

索引匹配原则b+树

索引查询优化

回表查询

事务

acid

隔离级别

### 3.1.2 数据库分类

#### 3.1.2.1 MySql

#### 3.1.2.2 Oracle

#### 3.1.2.3 Db2

#### 3.1.2.4 Postgre

## 3.2 JavaWeb

### 3.2.1 前端基础

#### 3.2.1.1 Html

|  |
| --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\1-前端基础\1-Html\HTML.xmind](3-Java-亿级流量技术栈/2-JavaWeb/1-前端基础/1-Html/HTML.xmind) |

#### 3.2.1.2 Css

|  |
| --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\1-前端基础\2-CSS\CSS.xmind](3-Java-亿级流量技术栈/2-JavaWeb/1-前端基础/2-CSS/CSS.xmind) |

#### 3.2.1.3 Javascript

##### 3.2.1.3.1 Js核心基础

|  |
| --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\1-前端基础\3-JavaScript\1-JS核心基础\JavaScript.xmind](3-Java-亿级流量技术栈/2-JavaWeb/1-前端基础/3-JavaScript/1-JS核心基础/JavaScript.xmind) |

##### 3.2.1.3.2 Js基础框架

###### 3.2.1.3.2.1 Ajax（js框架）

|  |
| --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\1-前端基础\3-JavaScript\2-JS基础框架\1-Ajax\Ajax.xmind](3-Java-亿级流量技术栈/2-JavaWeb/1-前端基础/3-JavaScript/2-JS基础框架/1-Ajax/Ajax.xmind) |

###### 3.2.1.3.2.2 Jquery（js框架）

|  |
| --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\1-前端基础\3-JavaScript\2-JS基础框架\2-Jquery\JQuery.xmind](3-Java-亿级流量技术栈/2-JavaWeb/1-前端基础/3-JavaScript/2-JS基础框架/2-Jquery/JQuery.xmind) |

##### 3.2.1.3.3 Js运行环境

### 3.2.2 前端技术栈

#### 3.2.2.1 ui框架

##### 3.2.2.1.1 JqueryUi

##### 3.2.2.1.2 EasyUi

##### 3.2.2.1.3 Bootstrap（CSS框架）

##### 3.2.2.1.4 Nui

##### 3.2.2.1.5 ElementUI

#### 3.2.2.2 S高级框架

##### 3.2.2.2.1 VUE(必会)

vue-resource过时

axios多，代替jquery

element ui

##### 3.2.2.2.2 React

##### 3.2.2.2.3 AgularJS

##### 3.2.2.2.4 Webpac

##### 3.2.2.2.5 Nodejs

##### 3.2.2.2.6 Wechat

### 3.2.3 javaweb核心基础

#### 3.2.3.1 Servlet基础

|  |  |
| --- | --- |
| 【笔记】 | [3-Java-亿级流量技术栈\2-JavaWeb\3-JavaWeb核心基础\1-Servlet基础\Servlet.xmind](3-Java-亿级流量技术栈/2-JavaWeb/3-JavaWeb核心基础/1-Servlet基础/Servlet.xmind) |
| 【图1】servlet执行原理图 |  |
| 【图2】servlet执行原理图 |  |

生命周期

listener

filter

cookie/session

sevlet框架

jsp（过时）

指令元素与动作元素

隐式对象

四大作用域

el表达式和jstl

执行流程与原理

web层的mvc开发模式

|  |  |
| --- | --- |
| ◎ [3-Java-亿级流量技术栈\2-JavaWeb\3-JavaWeb核心基础\1-Servlet基础\Servlet.xmind](3-Java-亿级流量技术栈/2-JavaWeb/3-JavaWeb核心基础/1-Servlet基础/Servlet.xmind) | |
| spring原理图 | <Java架构线路图.xmind> |
|  |  |
|  |  |

#### 3.2.3.2 JSP

### 3.2.4 web容器

#### 3.2.4.1 Tomcat

#### 3.2.4.2 Jetty

#### 3.2.4.3 Weblogic

#### 3.2.4.4 Websphere

### 3.2.5 混乱的java日志体系(排查问题)

#### 3.2.5.1 Log4j

#### 3.2.5.2 Jui

#### 3.2.5.3 Jcl

#### 3.2.5.4 Slf4j

#### 3.2.5.5 Logback

#### 3.2.5.6 Log4j2

## 3.3 JavaEE

### 3.3.1 MVC框架

#### 3.3.1.1 Struts2

#### 3.3.1.2 SpringMVC

基本使用

DispatcherServlet分析

### 3.3.2 模板引擎（至少会1-2种）

#### 3.3.2.1 Freemarker

#### 3.3.2.2 Volecity

#### 3.3.2.3 Thymleaf

### 3.3.3 Orm框架

#### 3.3.3.1 Hibernate

#### 3.3.3.2 Jpa

#### 3.3.3.2 Mybatis

轻量级orm框架，配置文件

代替原生jdbc完成crud

缓存和一对多，多对一映射

核心配置文件和Mapper

基于注解的方式

基于provider的方式

generator插件逆向生成

mabatis plus和通用mapper

pagehelper

整合spring

#### 3.3.3.3 SpringJDBC

### 3.3.4 Spring

基本使用

ioc/di

aop

事务

国际化

spring启动流程(高级源码)

bean的生命周期(高级源码)

spring的各种扩展点postProcessor

## 3.4 专题框架(针对某一个领域的框架)

### 3.4.1 Webservice(慢慢过时)

#### 3.4.1.1 Cxf

#### 3.4.1.2 Axis2

### 3.4.2 Http

#### 3.4.2.1 HttpClient

#### 3.4.2.2 Okhttp

#### 3.4.2.3 RestTemplate

#### 3.4.2.4 Jsoup

### 3.4.3 Xml

#### 3.4.3.1 Dom4j

#### 3.4.3.2 Jdom

#### 3.4.3.3 Xstream

#### 3.4.3.4 Jaxb

### 3.4.4 Json

#### 3.4.4.1 Fastjson

#### 3.4.4.2 Jackson

#### 3.4.4.3 Gson

#### 3.4.4.4 Json-lib

### 3.4.5 Excel操作

#### 3.4.5.1 Apache poi

#### 3.4.5.2 Excel4j

### 3.4.6 工作流

#### 3.4.6.1 Activty

#### 3.4.6.2 Jbpm

### 3.4.7 缓存

#### 3.4.7.1 Ehcache(多)

#### 3.4.7.2 Redis

#### 3.4.7.3 Memcache

#### 3.4.7.4 Oscache

#### 3.4.7.5 Guavacache

#### 3.4.7.6 LoadingCache

#### 3.4.7.7 MongoDB

### 3.4.8 权限与安全

#### 3.4.8.1 Shiro

#### 3.4.8.2 Springsecurity

#### 3.4.8.3 Jwt

#### 3.4.8.4 Opensso

#### 3.4.8.5 Auth2

### 3.4.9 定时任务

#### 3.4.9.1 原生Timer

#### 3.4.9.2 Quartz

#### 3.4.9.3 Scheduled(spring)

### 3.4.10 单元测试

#### 3.4.10.1 Junit单元测试

#### 3.4.10.2 Mock框架

### 3.4.11 Nosql

#### 3.4.11.1 Redis

#### 3.4.11.2 Mogodb

#### 3.4.11.3 Memcached

### 3.4.12 数据库连接池

#### 3.4.12.1 Dbcp

#### 3.4.12.2 C3p0

### 3.4.13 文档与调试

#### 3.4.13.1 Swagger2（生成api文档）

#### 3.4.13.2 Postman

## 3.5 微服务SpringBoot

starter

jdbc

jedis

mybatis

mq

cache

jest

..

零配置

自动注入原理

自定义starter

约定大于配置-目录结构（放东西到指定目录）

endpoint监控

## 3.6 分布式架构

### 3.6.1 主流分布式架构

有了分布式架构，会衍生出很多问题，所以衍生一堆分布式组件

#### 3.6.1.1 基于tcp的dubbo(和springcloud平分天下)

soa架构（面向服务的架构）

dubbo的注册中心

dubbo的负载均衡

dubbo admin控制台

spring中使用dubbo

#### 3.6.1.2 基于http/https协议的springcloud（下面小模块会1-2个）最火

##### 3.6.1.2.1 注册与发现中心

###### 3.6.1.2.1.1 Nacos

###### 3.6.1.2.1.2 Zookeeper

###### 3.6.1.2.1.3 Consul

###### 3.6.1.2.1.4 Eureka

###### 3.6.1.2.1.5 K8s

##### 3.6.1.2.2 配置中心

###### 3.6.1.2.2.1 Nacos

###### 3.6.1.2.2.2 Spring cloud config

###### 3.6.1.2.2.3 Apollo(携程提供)

##### 3.6.1.2.3 远程调用

###### 3.6.1.2.3.1 Openfeign

###### 3.6.1.2.3.2 Resttemplate

##### 3.6.1.2.4 负载均衡

###### 3.6.1.2.4.1 Ribbon

###### 3.6.1.2.4.2 Loadbanlance

##### 3.6.1.2.5 网关

###### 3.6.1.2.5.1 Spring gateway

###### 3.6.1.2.5.2 Zuul2(慢慢淘汰)

##### 3.6.1.2.6 降级，熔断，限流

###### 3.6.1.2.6.1 Hystrix

###### 3.6.1.2.6.2 Sentinel

##### 3.6.1.2.7 链路追踪

###### 3.6.1.2.7.1 Zipkin

###### 3.6.1.2.7.2 Skywalking

###### 3.6.1.2.7.3 Seluth

###### 3.6.1.2.7.4 Cat

### 3.6.2 分布式缓存

#### 3.6.2.1 redis

### 3.6.3 全文搜索引擎（大数据量mysql再怎么优化都不行）

#### 3.6.3.1 ES eleastic search

#### 3.6.3.2 Solr

### 3.6.4 elastic stack (el栈) es+logstash+kibana(ELK)现在慢慢被EFK替代，F就是beat

es

beat(其他日志采集工具如fluentd) + logstash

kibana

solr

### 3.6.5 分库分表

#### 3.6.5.1 Mycat

#### 3.6.5.2 Sharding-jdbc

#### 3.6.5.3 Sharding-sphere

#### 3.6.5.4 数据库灾备

### 3.6.6 分布式任务

#### 3.6.6.1 Xxl-job

### 3.6.7 分布式事务

跨库事务

分布式服务事务解决方案

分布式一致性

2PC、TCC

最大努力通知方案

saga

### 3.6.8 消息队列mq

#### 3.6.8.1 Jms

#### 3.6.8.2 Amqp

#### 3.6.8.3 Activemq

#### 3.6.8.4 Rocketmq

#### 3.6.8.5 Rabbitmq

#### 3.6.8.6 Kafka

### 3.6.9 分布式链路追踪

#### 3.6.9.1 Zipkin

#### 3.6.9.2 Skywalking

### 3.6.10 分布式锁

#### 3.6.10.1 Redis

#### 3.6.10.2 Zookeeper

#### 3.6.10.3 Mysql

### 3.6.11 分布式对象存储(图片文档js等)

#### 3.6.11.1 Oss

#### 3.6.11.2 Aws s3

#### 3.6.11.3 Fastdfs

#### 3.6.11.4 Cdn

### 3.6.12 负载均衡

#### 3.6.12.1 Nginx

Tengine

Openresty

#### 3.6.12.2 Haproxy

#### 3.6.12.3 LVS

#### 3.6.12.4 KeepAlived

#### 3.6.12.15 Zookeeper

## 3.7 三方源码

## 3.8 测试技术

LoadRunner