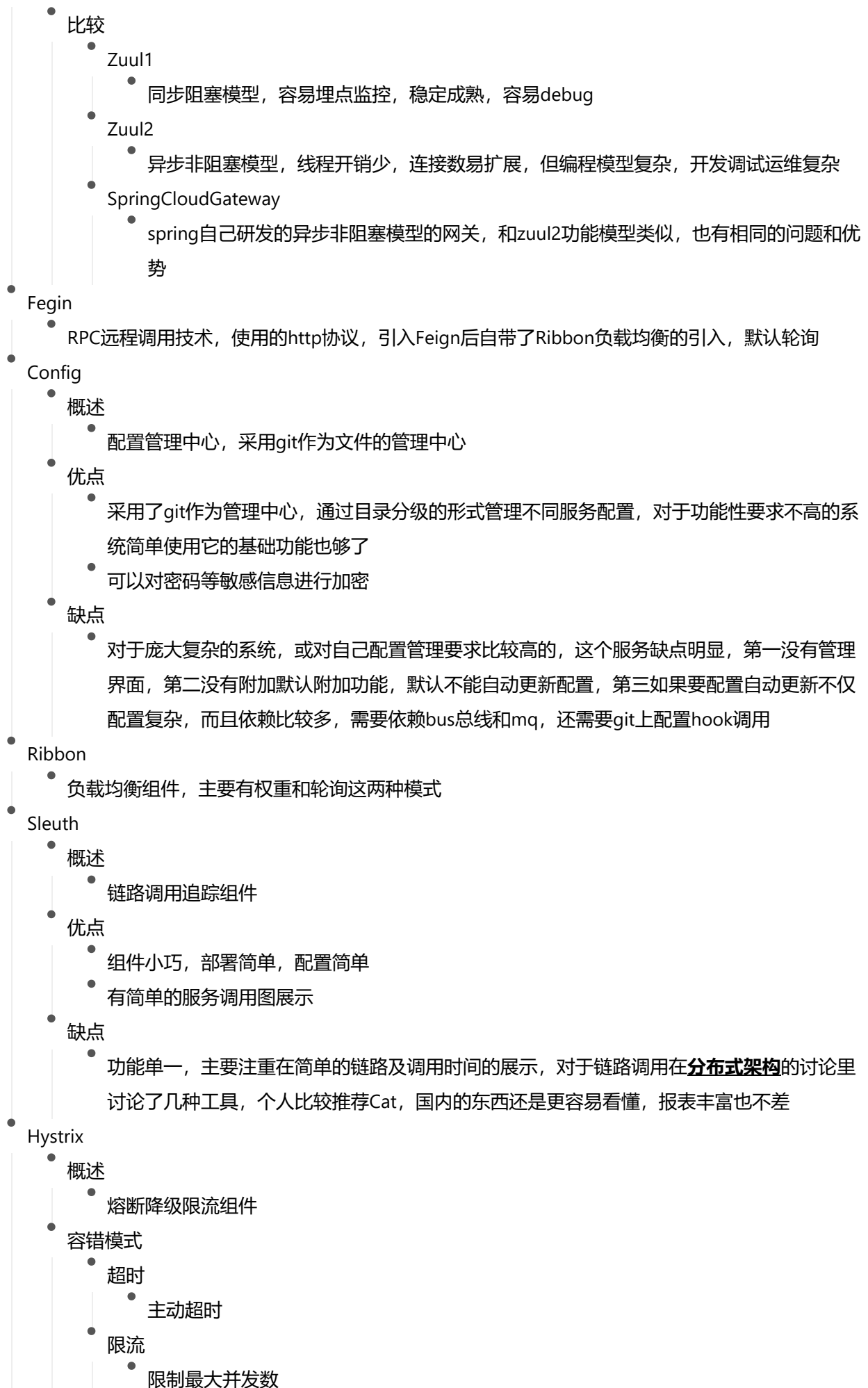
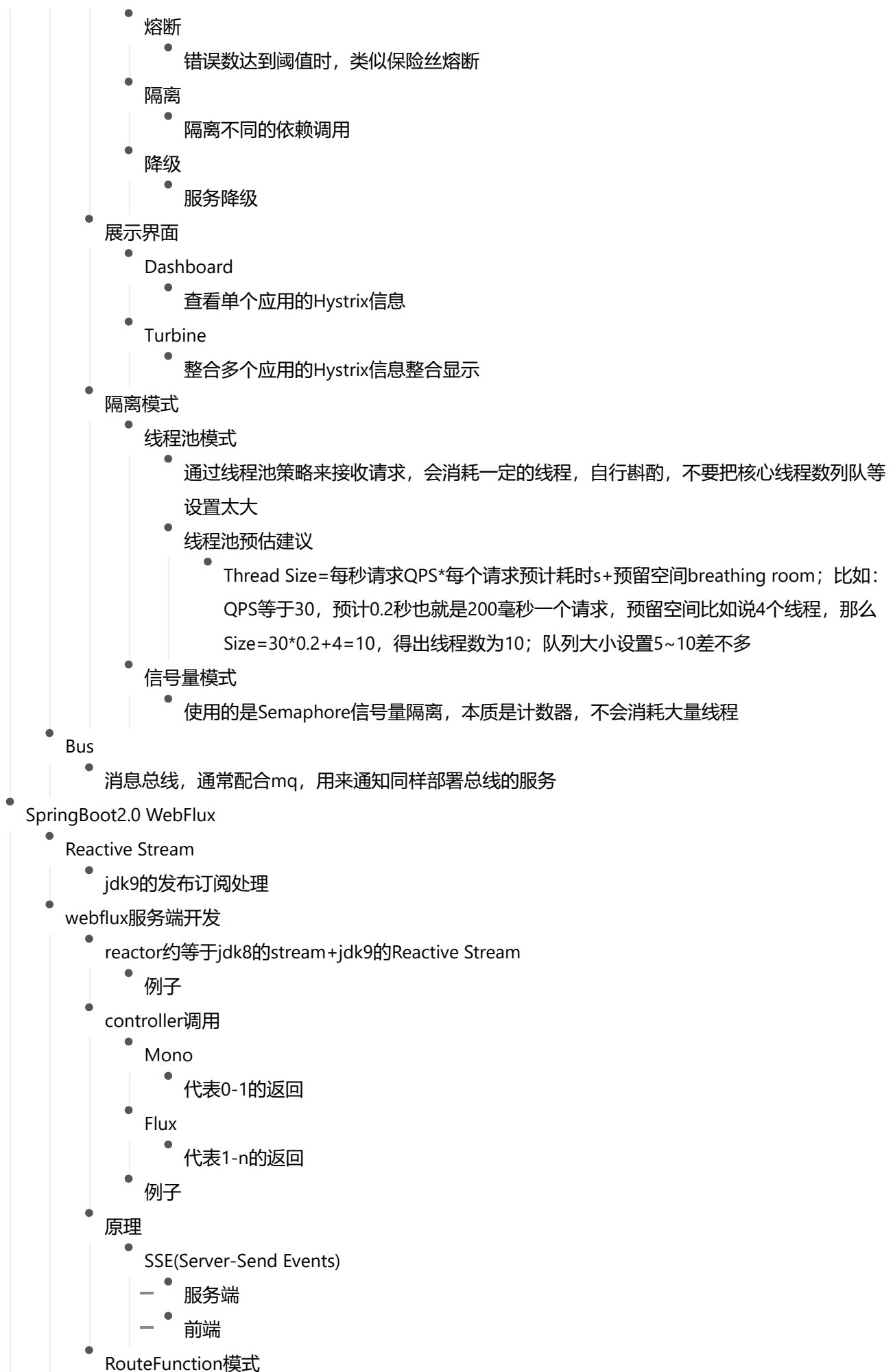
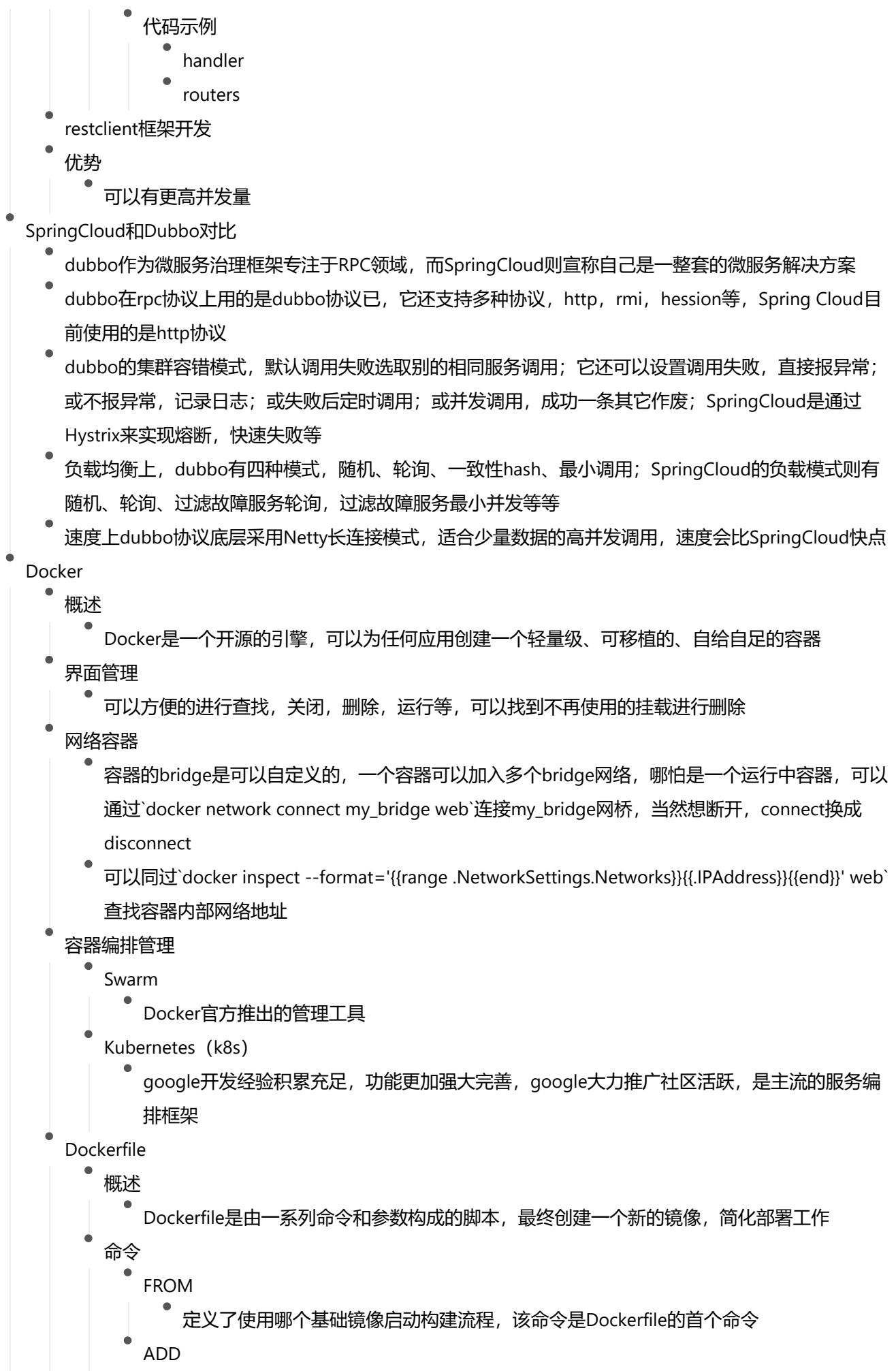


Java整体知识架构详解-之微服务架构

- SpringBoot
 - 概述
 - 实现了自动配置，集成了大量常用第三方库，开箱即用
 - spring-boot-starter
 - 将代码改造成starter可插拔插件，做到引入pom后开箱即用
 - SpringBoot可直接集成的常用服务
 - 常用安全框架Security，可用于登录认证，OAuth2授权认证等
 - 常用数据库mysql，oracle等快速集成
 - 常用Nosql数据库，redis，MongoDB，Solr，Elasticsearch，InfluxDB等快速集成
 - 常用缓存架构ehcache2.x，Redis等
 - 常用消息服务，JMS（ActiveMQ等），AMQP（Rabbitmq等），Kafka相关mq支持
 - 常用定时任务Quartz Scheduler
- SpringCloud
 - Eureka
 - 注册中心，管理协调分布式集群信息，类似的有zookeeper，Nacos等
 - Eureka1.x目前还在维护，但预计的2.x已经宣布不再开发
 - Zuul
 - 概述
 - 网关服务
 - 网关基本功能
 - 单点入口
 - 路由转发
 - 熔断限流
 - 日志监控
 - 安全认证
 - 网关高级应用
 - 红绿部署
 - 开发者测试分支
 - 埋点测试
 - 压力测试
 - 调试路由
 - 金丝雀测试
 - 粘性金丝雀
 - 失败注入测试
 - 降级测试







- 命令有两个参数，源和目标，作用是从源系统的文件系统上**复制文件**到目标容器的文件系统，如果源是个URL，则会下载内容并复制到容器中
- RUN
 - RUN命令是Dockerfile执行命令的核心部分，它接收命令作为参数并用于创建镜像，比如`RUN aptitude install -y riak`
- CMD
 - 和RUN命令相似，CMD可以用于执行特定的命令，和RUN不同的是，这些命令不是在构建镜像的过程中执行的，而是用在**镜像构建容器后被调用**，用法：CMD application "argument", "argument", .
- ENTRYPOINT
 - 配置容器启动后执行的命令，并且**不可被docker run提供的参数覆盖，每个Dockerfile只能有一个ENTRYPOINT**，指定多个只有最后一个生效；可以从CMD中移除`application`仅仅保留参数，参数会自动传递给ENTRYPOINT。例如：ENTRYPOINT echo CMD "Hello world"
- ENV
 - 用于设置环境变量。例如：ENV SERVER_WORKS 4
- EXPOSE
 - 用来指定端口，使容器内应用可以通过端口和外界交互。例如：EXPOSE 8080
- MAINTAINER
 - 一般放在Dockerfile起始部分FROM之后，用于声明作者。例如：MAINTAINER authors_name
- USER
 - 用于设置运行容器的UID。例如：USER 751
- VOLUME
 - 用于让容器访问宿主主机上的目录。例如：VOLUME ["/dir_1","/dir_2" ..]
- WORKDIR
 - 用于指明CMD命令运行的目录。例如 WORKDIR /path
- Dockerfile构建镜像
 - docker build -t 镜像名称 . (最后还有个点表示以此目录中的Dockerfile构建镜像)
- Docker Compose
 - 概述
 - 用于定义和运行多容器Docker的应用程序工具，需要安装Docker Compose
 - Docker Compose常用命令
 - docker-compose ps
 - 列出所有容器
 - docker-compose logs
 - 查看服务日志输出
 - docker-compose port eureka 8761
 - 打印绑定的公共端口
 - docker-compose build

- 构建或重新构建服务
- docker-compose start eureka
 - 启动指定服务已存在的容器
- docker-compose stop eureka
 - 停止已运行的服务的容器
- docker-compse rm eureka
 - 删除指定服务的容器
- docker-compse up
 - 构建、启动容器
- docker-compose scale user=3 movie=3
 - 设置指定服务运行容器个数
- docker-compose run web bash
 - 在一个服务上执行一个命令
- docker-compose.yml 属性
 - version
 - 指定docker-compse.yml文件的写法格式
 - 指定docker-compse.yml文件的写法格式
 - 指定docker-compse.yml文件的写法格式
 - version
 - image
 - 指定服务所使用的镜像
 - services
 - 多个容器集合
 - 多个容器集合
 - 多个容器集合
 - services
 - build
 - 用于构建镜像的参数指定, 可以是一个路径, 也可以是一个对象
 - 用于构建镜像的参数指定, 可以是一个路径, 也可以是一个对象
 - 用于构建镜像的参数指定, 可以是一个路径, 也可以是一个对象
 - build
 - volumes
 - 卷挂载路径
 - volumes_from
 - 从另一个服务或容器挂载它的所有卷
 - expose
 - 暴露端口给连接的服务, 而不暴露给主机
 - ports
 - 对外暴露的端口定义
 - links

- 将指定容器连接到当前连接，可以设置别名，避免ip方式导致的容器重启动态改变的无法连接情况
- external_links
 - 链接到外部容器
- network_mode
 - 设置网络模式
- command
 - 覆盖容器启动后默认执行的命令
- dns
 - 配置dns服务器，可以是一个值或列表
- pid
 - 跟主机系统共享进程命名空间。打开该选项的容器可以相互通过进程 ID 来访问和操作。
- docker run支持的选项
 - working_dir, entrypoint, user, hostname, domainname, mem_limit, privileged, restart, stdin_open, tty, cpu_shares
- dns_search
 - 配置DNS搜索域，可以是一个值或列表
- environment
 - 环境变量配置，可以用数组或字典两种方式
- env_file
 - 从文件中获取环境变量，可以指定一个文件路径或路径列表，**优先级低于environment**指定的环境变量
- 最佳实践
 - 更新容器
 - 当服务的配置发生更改时，可使用docker-compse up命令更新配置，此时Compose会删除旧容器并创建新容器
 - 利用好links
 - 服务之间相互访问用links定义别名，从而使用该别名访问其它服务